A Data Set of Internet Claims and Comparison of their Sentiments with Credibility

Amey Parundekar
Vellore Institute of Technology
Chennai, India
parundekaramey@gmail.com

Dr. Susan Elias
Vellore Institute of Technology
Chennai, India
susan.elias@vit.ac.in

Dr. Ashwin Ashok
Georgia State University
Atlanta, USA
aashok@gsu.edu

ABSTRACT

In this modern era, communication has become faster and easier. This means fallacious information can spread as fast as reality. Considering the damage that fake news kindles on the psychology of people and the fact that such news proliferates faster than truth [11], we need to study the phenomenon that helps spread fake news. An unbiased data set that depends on reality for rating news is necessary to construct predictive models for its classification. This paper describes the methodology to create such a data set. We collect our data from snopes.com which is a fact checking organisation. Furthermore, we intend to create this data set not only for classification of the news but also to find patterns that reason the intent behind misinformation. We also formally define an *Internet Claim*, its credibility, and *sentiment* behind such a claim. We try to realize the relationship between the sentiment of a claim with its credibility. This relationship pours light on the bigger picture behind the propagation of misinformation. We pave the way for further research based on the methodology described in this paper to create the data set and usage of predictive modeling along with research based on psychology/mentality of people to understand why fake news spreads much faster than reality.

CCS CONCEPTS

• **Information systems** → *Retrieval models and ranking*; *Retrieval models and ranking*; • **Computing methodologies** → **Natural language processing**; **Information extraction**.

KEYWORDS

datasets, natural language processing, web mining, sentimental analysis, credibility rating

ACM Reference Format:
Amey Parundekar, Dr. Susan Elias, and Dr. Ashwin Ashok. 2019. A Data Set of Internet Claims and Comparison of their Sentiments with Credibility. In *Anchorage '19: The first workshop on Truth Discovery and Fact Checking: Theory and Practice, August 05 2019, Anchorage, AK, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Anchorage '19, August 05 2019, Anchorage, AK, USA
© 2019 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

1.1 Purpose

Since the dawn of the internet, it has become increasingly easier to communicate information to anyone in this world at any time. Today, any piece of information travels faster over the world than it ever did. Advancements in technology have enabled us to reach an increasingly higher number of people in lower amounts of time. Although this proves to be a boon to society, it has several drawbacks too. It is paramount to broadcast critically essential information, related to food supplies and medical help, during a natural disaster and current technology does a perfect job at doing so. But, this technology also helps people spread fake news around the world that might at times critically affect the way people behave. Hence it has become necessary to sift between content over the internet to separate truth from the mendacious. An article recently published by Scientific American [2] stated that, lately, social media has been the top source for news, for the majority of people in a country like the USA among others.

In this paper we describe the development of a data set, needed for research, aimed at analyzing random claims made by people over various social media and their credit rating as rated by *snopes.com*. It further aims to compare the sentiment of these claim with their credibility. This comparison helps us analyse the intent behind propagation of misinformation. *snopes.com* is one of the many fact-checking sources over the internet and we have used it to create our data set. The data set consists of internet claims, their ratings, sentimental analysis of the ratings, the origin of the claims and analysis of the claims in general. We not only take *true* and *false* as labels/ratings into consideration but also consider some other labels/ratings like *mis-attribution* or *mis-captioned* in this data set. This paper aims at discussing simple methods to extract such data over the internet in a time and cost effective way and finding a relation between the sentiment of such claims and either their truthfulness or falsehood.

1.2 Challenges and Proposed Approach

One of the major challenges in the classification of news is the identification of credible sources that act as reference classifiers. These, mainly, are fact-checking organizations. The verisimilitude of misinformation makes it very difficult for such classifiers and hence any predictive models to draw a line between reality and fiction. There are several factors involved, that need to be analyzed before we avow any piece of information as truth. One of the major factor, that often gets neglected, and thus pose a major challenge on current news classifiers is the **temporal**. A piece of information

might hold at a particular time and yet at certain other times regarded fallacious. Another such important factor is the **sentiment** behind the news. A piece of information can either please everyone or infuriate them thereby causing overall conflict in the society. Hence, the real challenge faced by current day misinformation classifiers is the appropriate incorporation of these factors into their data sets and learning models.

For the collection of data that results in the creation of such a data set we propose an approach that first formally defines all things necessary for maintaining the authenticity of this data set. After doing so, we use a combination of web mining and natural language processing to finally generate the data set that pulls data from online fact-checking organizations. We also propose to incorporate temporal and sentimental factors in this data set. Hence instead of being a binary classification, we build a n-nary data set, where n can depend on the several factors that we account into the description of data. This way, we account for the temporal factor. Information, besides being ***true*** and ***false*** can also be ***outdated***, which means, in present time, the analysis of truth or fallacy of that piece of information has become irrelevant.

2 RELATED WORK

There have been several efforts in the past to analyze fake news on social media. Most of these sources consider fake news detection as a binary classification problem. Particularly a paper on data mining perspective of fake news published by the Computer Science and Engineering department of the Arizona State University, Tempe [9] defines fake news as *a news article that is intentionally and verifiable false*. They also define the prediction function for fake news detection as:

$$\mathcal{F}(a) = \begin{cases} 1, \text{ if } a \text{ is a piece of fake news,} \\ 0, \text{ otherwise.} \end{cases}$$

where a is a news article and $\mathcal{F}$ is the prediction function that we want to learn. This definition seems quite apt for building a simple binary classifier but in further sections, we will build upon and extend this definition to support a multi-class classification model to supplement our data set.

Several efforts have also been made in the past for the creation of data sets that complement research based on the detection of fake news. One such comprehensive data set is the CREDBANK [4] data set which is a big corpus of Tweets and their credit ratings as assessed by 30 Amazon Mechanical Turks. CREDBANK's creators thus created a good blend of manpower and computation to create their data set. A sample of their data set as provided on their GitHub page looks as shown in Table 1.

Here as we see, we have Tweets tokenized after removal of stopping words and presented in the table as tokens of keywords and not directly as tweets. In the Cred_Ratings and Reason columns of the table, we see a list of ratings of credibility, rated on a scale from −2 to 2 in the prior column and respective reason for that particular rating in the next. Each entry in this list of ratings and reasons is representing an Amazon Mechanical Turk, providing his/her own rating and their reason for that rating. Researchers at the Indiana University Observatory on Social Media have made several strides in the field of fake news detection as well. In doing so, they have launched several applications that study fake news. Hoaxy [5] is a good example of their work. As their website describes, "Hoaxy is a search engine that shows users how stories from low-credibility sources spread on Twitter."

Table 1: CREDBANK SAMPLE DATA

topic_key	topic_terms	Cred_Ratings	Reason
louis_ebola_...	[u'louis', uébola; uńursé]	[1́; -́1; 2́; -́2; 0́; 2́; 0́;....]	[Ńurses union describes the procedures taken by nurse who now has Ebola from treating a patient., ...]

Figure 1, shows a sample output of what Hoaxy is capable of. This is a plot for the search query: "Women in Sweden are paid to marry immigrants".

Another study at MIT, analyzed all Tweets from 2006 to 2017 to find patterns among them. They stated that false news spreads significantly farther, faster, deeper, and more broadly than the truth. [11]

3 BACKGROUND

The studies in the previous section and their results, suggests that it is necessary to study the sentiment behind false claims to understand the bigger picture. Some people spread false news on purpose but some people seem to spread such news unknowingly and without verification. The sentiment of a claim sheds a light on why people knowingly/unknowingly end up making/sharing false claims faster, deeper and farther than true claims. We start by formally defining an *Internet Claim*, *Fact-Checking function* for a claim and *Sentiment* of a claim.

3.1 Internet Claim

An **Internet Claim** is any piece of information published/written or has any other form of presence over the internet visible to all on any media by a person or an entity that might be true or false.

Only after proper examination of such a claim and it's fact-checking, should we make any conclusion about the authenticity of such a claim or form an opinion about it. Notice that the word "should" here is very important as it highlights the fact that we "can" form opinions but "should not" do so without verification. This might be a restatement but it is very essential for understanding the observed phenomenon of faster false news travel.

3.2 Fact-Checking Function

Fact-Checking Function for an Internet Claim is a function that rates the credibility of the claim. Notice that this is ***not*** a **predictive function** like the one given in Equation 2 but rather an **assertive function** that rates the credibility of an Internet Claim without any sort of learning or prediction. The structure and output of this function is based possibly on ground truth and reality which is what we desire but it's sometimes also based on belief. One possible but not limiting mathematical model of this function is:

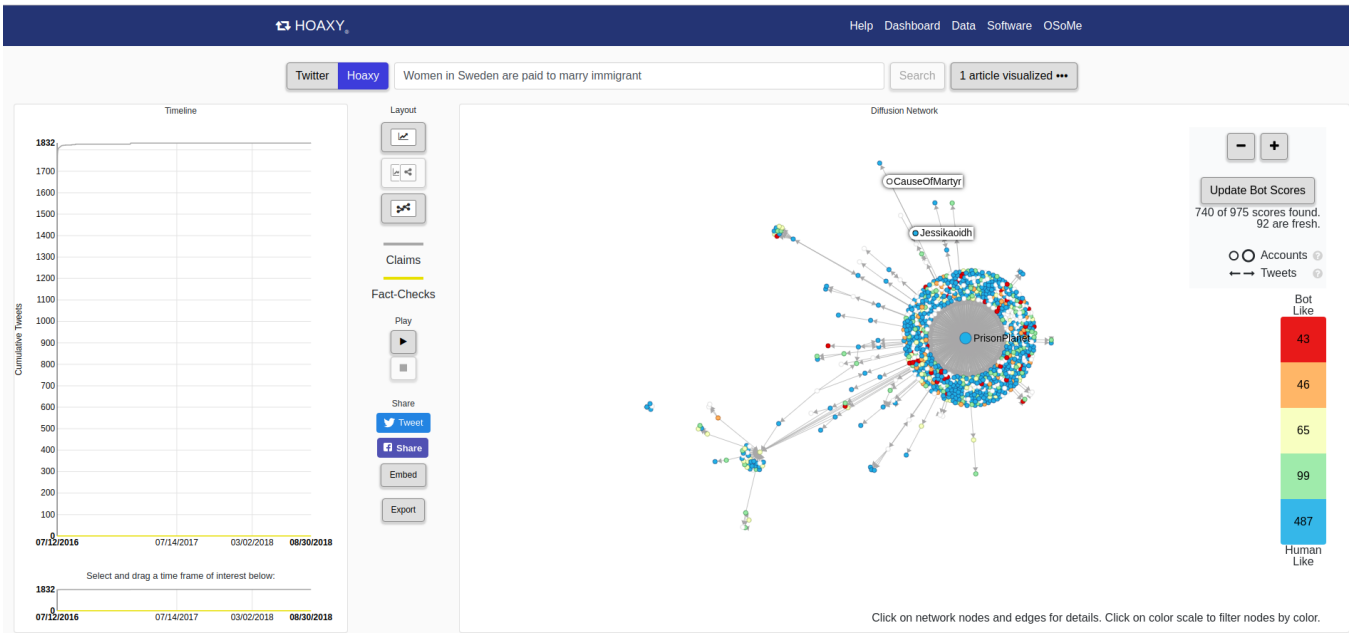

Figure 1: Example of Hoaxy's network plot.

$$\mathcal{S}(a) = \begin{cases} \text{False, if } a \text{ is a piece of fake news,} \\ \text{True, otherwise.} \end{cases}$$

where a is a news article and $\mathcal{S}$ is an assertion function that rates the credibility of claims.

Wait, maths and beliefs? What are we talking about here? Right? Well, to give an example, "beliefs" are very much like the postulates we see and agree to be true without any proof in Euclidean Geometry. It is important to notice that, instead of thinking of this as a mathematical function we need to think of the Fact-Checking Function as a fact-checking organization like snopes.com. Of course, goes without saying that a Fact-Checking Function should, assert a truthful output, failing to which we would not be sure if our credibility ratings for a claim are correct or not. This, in fact, means that we need to assess the credibility of a Fact-Checking Function, which for our examples means that we need to verify that snopes.com is a credible source and does not provide us with wrong ratings for news.

As we observe, we face a challenge where we have to apply another Fact-Checking Function on the initial Fact-Checking Function to rate the credibility of the Fact-Checking Function itself so that we know for sure that our ratings are correct or not. Notice that of course, the two Fact-Checking Functions here should be different from each other and not the same.

$$\mathcal{S}_2(\mathcal{S}_1(a)) = \begin{cases} \text{False, if } \mathcal{S}_1 \text{ is source of false ratings,} \\ \text{True, otherwise.} \end{cases}$$

where a is a news article and S_1 is the initial credibility assertion function that rates the credibility of the claims and S_2 is the credibility assertion function for the initial function. To overcome this conundrum, we define something called the Event function.

3.3 Event Function

Event Function for an Internet Claim is the function that is based on reality and not on belief. It tells us what actually happened. It gives a binary output of either True if the event described in a claim actually happened or False if the event described in a claim never happened in this reality. Notice that such a binary behavior is not necessarily expected by a Fact-Checking Function. A Fact-Checking Function can be ternary or have even higher orders. (with example states like True, False, Mostly True, Mostly False, Mis-captioned, etc.).

$$\mathcal{E}(a) = \begin{cases} \textbf{True}, \text{ if } a \text{ is a is piece of claim that describes} \\ \qquad\qquad \text{an event that actually occurred in reality,} \\ \textbf{False}, \text{ otherwise.} \end{cases}$$

where a is a news article and $\mathcal{E}$ is an assertion function that rates the credibility of claims.

We thus say that any Fact-Checking Function that is derived from/based upon the Event Function correctly assess the credibility of an Internet Claim and then that function and any Fact-Checking Function(s) derived from it can be considered as credible rating sources. This is like saying that a representative from snopes.com actually went on to look for physical details of an event and found conclusive proofs in favor of the event or against it or they derived their results from some other fact-checking organizations which found such conclusive proofs which made both of their information credible.

After sifting through all resources available online, to collect data related to internet claims, we chose snopes.com as it turned out to be a very good source to collect the data for several interesting reasons. On observing snopes.com's Twitter account through the Twitter API [15], it was found that the account had tweets of *Internet Claims*, with every tweet containing a link to their website, which

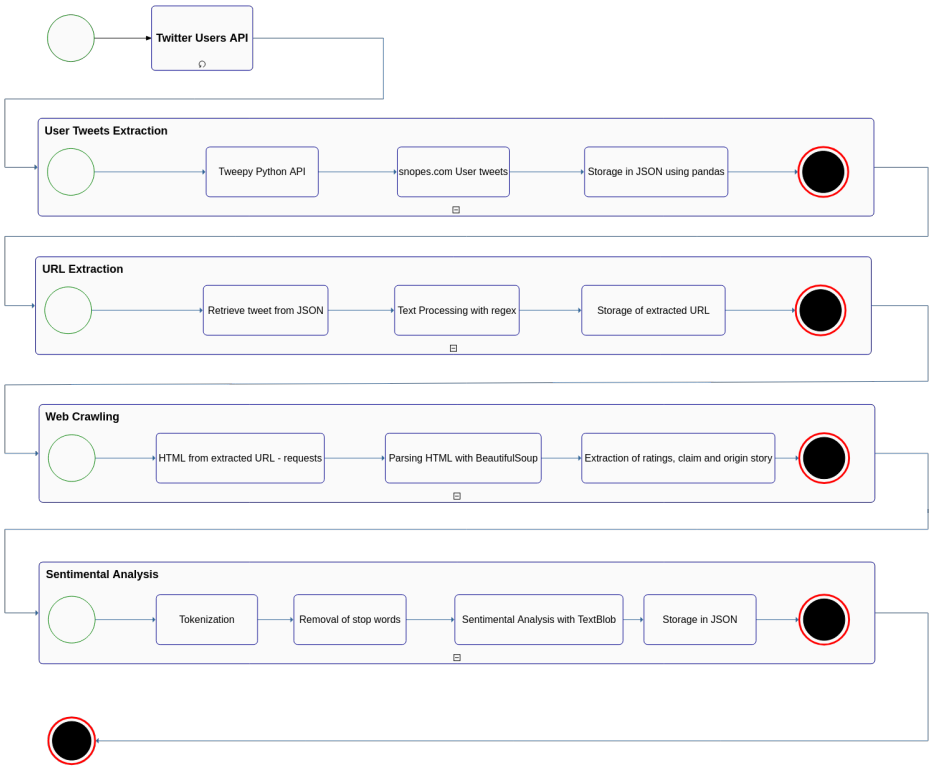

Figure 2: Process flow of data set creation

had that claim from the internet analyzed on being True, False, Mis-captioned, etc. along with proper reasoning and comparison with the ground truth to verify their rating for the claim provided as "Origin". This makes snopes.com a Fact-Checking Function that is dependent directly on the Event Function hence making their credibility ratings credible.

4 SENTIMENT OF A CLAIM

The sentiment of an Internet Claim is the emotional bias of the claim. It can mainly be classified into Positive, Negative and Neutral. To put it in common terms, the Sentiment of a claim is how you feel about the claim when you read it. If it sparks off anger, then the claim has a negative sentiment. Whereas if it fosters joy or courage or happiness, the claim has a positive sentiment. And if neither happens then the claim has a neutral sentiment. For a article a, the *sentiment* can be defined as:

$$\mathcal{B}(a)\begin{cases} > 0, \text{ if positive} \\ = 0, \text{ if neutral,} \\ < 0, \text{ if negative} \end{cases}$$

4.1 Getting Data

The process followed by us for creation of the data set is given in the Process Flow diagram 2. The code to get data with Twitter API was written in python and inspired by Vincent Russo's GitHub repository [8] on working with tweepy [13] which is a python library for the Twitter Developer's API. The code was written to collect user tweets of snopes.com's Twitter account for the reason as mentioned in section 3.3. According to Twitter's API limitation's one can extract 15 pages worth tweets with 200 tweets per page from a user's profile. Keeping this in mind, tweets were extracted from snopes.com's profile multiple times with enough time gap between two collections, making sure that we don't collect already collected tweets. These tweets were stored in 15 separate files for each page, with 200 tweets in each file in JSON format. Sample tweet data is as follows:


```
"8": {
        "tweets": "Was Bill O'Reilly
                   found dead at his
                   Long Island home?
          https:\/\/t.co\/SGwagACMbW
         https:\/\/t.co\/Ppx1FhJeMm",
        "id": 1075020507186126853,
        "len": 101,
        "date": 1545139836000,
        "source": "AgoraPulse Manager",
        "likes": 4,
        "retweets": 2,
        "time": 1545139836000,
        "geo": null,
```

```
        "sentiment": -1,
        "token_list": [
            "Was",
            "Bill",
            "O",
            "Reilly",
            "found",
            "dead",
            "Long",
            "Island",
            "home"
        ]
    }
```


This is the eighth tweet from the first page's output file. Among the several available parameters provided by twitter [14] the one seen in the above example were used. Pandas [6] was used to deal with data frames used to handle the above data.

4.2 URL Extraction

As it is observed, the "tweets" section which contains the text from the actual tweet, contains a URL which maps to the corresponding post on snopes.com for the claim mentioned in that tweet. Hence, "*https:t.coSGwagACMbW*" in fact maps to "*https://www.snopes.com/fact-check/bill-oreilly-found-dead/*".

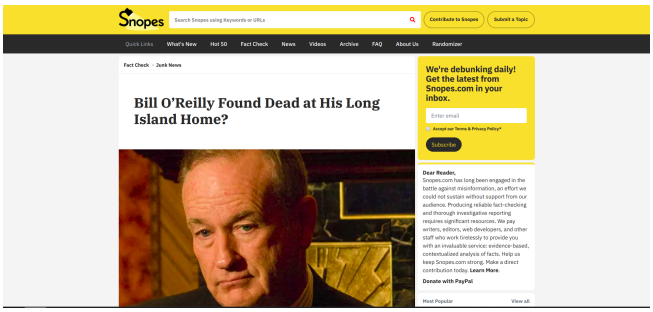

Figure 3: snopes.com/fact-check/bill-oreilly-found-dead/

The page in Figure 3 contains 3 main sections which are: ***Claim***, ***Rating*** and ***Origin***. The claim section contains the Internet Claim, the rating section contains the rating of this claim and the origin section gives an explanation about the origin of this claim. So our job amounts to extracting these URLs from the tweets of snopes.com's Twitter account. The most straight forward and elegant solution to extract these URLs is via the use of regular expressions. We used the following regular expression:

```
((?:(https?|s?ftp):\/\/)?(?:www\.)?
((?:(?:[A-Z0-9][a-zA-Z0-9-]{0,61}
[A-Z0-9]*\.)+)([A-Z]{2,6})|(?:\d
{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}))
(?::(\d{1,5}))?(?:(\/\S+)*))
```

This regular expression extracts all URLs from a text which contain http, https or ftp. It takes care of presence or absence of **www**, and the length of the URL. For further explanation on how this regular expressions works, one can have a look at the regex link [7] that we created. A few examples of how the regular expression works is given in Figure 4.

Figure 4: URL Extractor

4.3 Web Crawling

After extraction of the URLs from the snopes.com's profile's tweets, we get the HTML response of their web pages which have claims and ratings. Our work, then, amounts to finding the sections of the page which provide us with the claim and the rating information for that claim. After obtaining the ratings for the claim, it is stored back into the tweet's JSON data. We make use of BeautifulSoup [3] which is a python library to parse HTML and XML content, to extract this data from the URL's HTML response. A sample of the rated tweets or claims is shown next.


```
 "8": {
"origin-html": "[<div class=\"post-
body-card post-card card\">\\n<h3
class=\"card-header\"> Origin</h3>
\\n<div class=\"card-body\">\\n<p>
On 21 May 2017, the Daily USA
Update web site published
an article purporting to reveal
\\u201cmore details about
the sad death\\u201d of
former Fox News anchor
Bill O\\u2019Reilly:
</p>\\n<blockquote><p>
The Islip Coroner\\u2019s
Office stated that
last night,...]"
"token_list": [
  "Was" ,
  "Bill" ,
  "O" ,
  "Reilly" ,
  "found" ,
  "dead" ,
  "Long" ,
  "Island" ,
  "home"
] ,
"source": "AgoraPulse Manager" ,
"len": 101 ,
"claim-html": "[<p class=\"claim\">
```

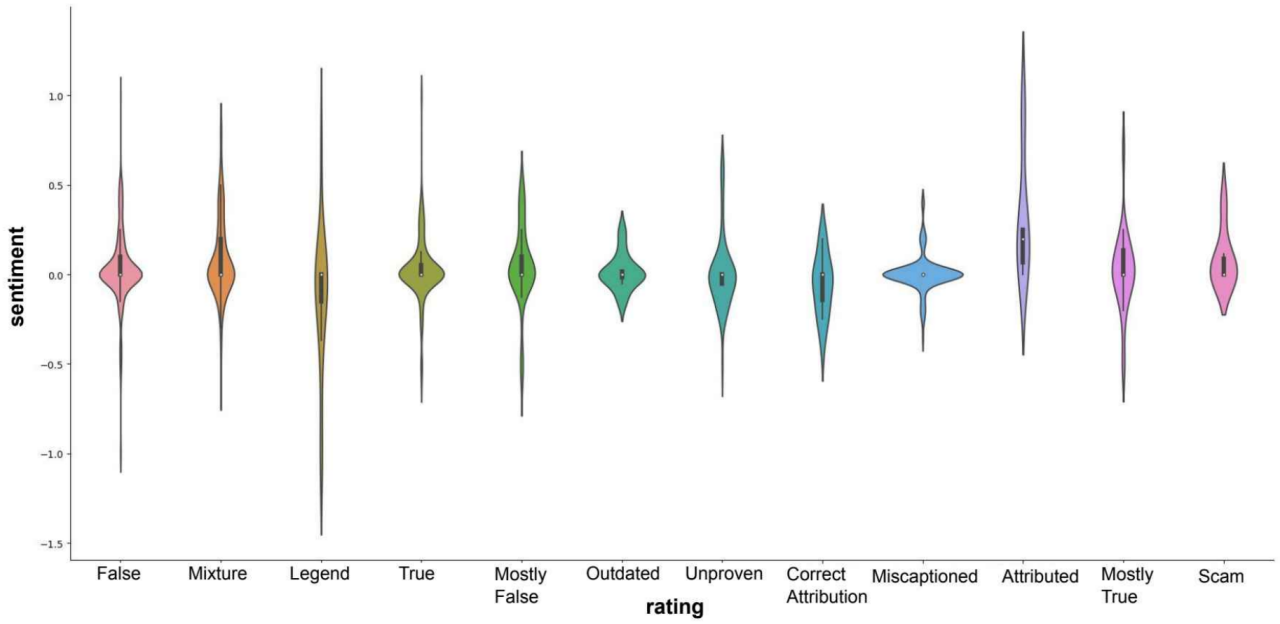

Figure 5: Violin plot of Sentiment vs Ratings

```
                Former Fox News host
                Bill O'Reilly
                was found dead on
                Long Island.</p>]" ,
  "date": 1545139836000 ,
  "rating-html": "[<span class=
  \"rating-name
  rating-label-false\">False</span>]" ,
  "likes": 4 ,
  "time": 1545139836000 ,
  "tweets": "Was Bill O'Reilly
  found dead at his
  Long Island home?
  https://t.co/SGwagACMbW
  https://t.co/Ppx1FhJeMm" ,
  "geo": null ,
  "id": 1075020507186126853 ,
  "retweets": 2
} ,
```

We have three new additions to the initial JSON data. *"origin-html"*, *"rating-html"* and *"claim-html"*. Hence we were successfully able to rate the claim and also find its origin story!

4.4 Sentiment Analysis of the Rated Claims.

TextBlob [12] is a python library used for text processing. It contains APIs that take in a string of text and return the overall sentiment expressed by the words in that string. It rates the sentiment between −1 to 1 and follows the logic we developed for sentiment in Section 3.3. Hence we get a normalized sentiment analysis of our claims via this python library.

4.5 Assembly of Data Set

We finally gathered together all the claims, their ratings, their origin story, the sentiment of these claims and nicely bundle them into a comma separated values. In doing so we get rid of all the claims that we couldn't find any rating information for. Example of the csv output and how our data set looks like is given below.

```
[Former Fox News
host Bill O'Reilly was found dead on
Long Island.], False,
-0.083333333333333,
[On 21 May 2017,
the Daily USA Update web site
published an article purporting to
reveal âĂmore details about the sad
deathâĂİ of former Fox News anchor
Bill OâĂŹReilly:...]
```

The first value is the claim, second value - "False" is the rating of the claim, the third value is the sentiment of the claim. And as we can see, it has a negative sentiment and the fourth value is the origin story. You can find this data set at the GitHub page [1] and have a glimpse of how it looks like in Table 2.

Finding that many tweets that snopes.com's Twitter account posted contained fact-checked internet claims and a link to their site containing detailed analysis of these internet claims was quite serendipitous. Without making this trivial discovery, we would not have been able to generate the data set. Notice that Twitter acted only as a passive medium that allowed us to mine the fact-checked internet claims from snopes.com. We did not in any way fact-check tweets but in fact used a "hack" we found in one of the fact-checking organisation's twitter account to our benefit. The data collected for

Table 2: iClaimNet DATASET
www.github.com/the-lost-explorer/iClaimNet

claim	rating	sentiment	origin
[Former Fox News host Bill O'Reilly was found dead on Long Island.]	FALSE	-0.083333333	[On 21 May 2017,the Daily USA Update web site published an article purporting to reveal Ãć more details about the sad death of former Fox News anchor Bill O'Reilly..]

the data set generated in the process described here, was collected from the most recent tweets of snopes.com's twitter account, which resulted in collection of the corresponding internet claims. Since the order in which snopes.com posted the internet claims on twitter was random, in the sense that even though it may or may not have been temporally serial, it formed no traceable patterns among itself, and every internet claim collected in such way was different from every other, the data collected was unrelated, mutually exclusive and hence formed a random pool of internet claims.

5 OBSERVATIONS

With our *comma separated values* of claims, ratings and their sentiments, one can start visualizing what is the sentimental trend behind a particular rating. We plot the sentimental value of a claim to its rating as seen in Figure 4. We have a *Violin Plot* of the ratings vs sentiment for every claim. On the X axis we have 12 different rating types—***True, False, Mostly True, Mostly False, Outdated, Miscaptioned, Mis-attributed, Unproven, Mixture, Legend, Scam, Correct Attribution***. A description of what each rating represents is given on snopes.com's rating page [10]. We demonstrate our data using a violin plot. A violin plot is a method of plotting numeric data. It is similar to a box plot, with the addition of a rotated kernel density plot on each side. Violin plots are similar to box plots, except that they also show the probability density of the data at different values, usually smoothed by a kernel density estimator. The violin plot of Figure 5 depicts the standard distribution, inter-quartile range and median of the sentiment score for each rating.

Here we can further cluster "False", "Mostly False", "Mis-attributed", "Mis-captioned" and "Scam" claims under a single category and "True", "Mostly True" and "Correct Attribution" under another single category due to a similarity in the meaning they express. The violin plot after such clustering is given in Figure 6.

The statics among the 1669 analyzed claims, is as follows:

```
Total False Claims: 495
Total True Claims: 160
False Positives: 306
False Negatives: 189
True Positives: 99
True Negatives: 61
```

Observation 1: Upon discussion in the section on assembly of data set we understood that the data set forms a random pool of internet claims and in such a random sample of about 1600 internet claims, the number of false claims is much higher than the number of true claims.

Observation 2: Moreover 38.18% of the false claims had a negative sentiment whereas 38.12% of the true claims had a negative sentiment. These two figures are almost the same.

Observation 3: When we look at the distribution graph of the sentiments(see Figure 6) we observe that claims that have a very high negative sentiment are false whereas, the claims that have a very high positive sentiment can be both true or false. This can also be verified by the figures. When we compare the false and true claims, we find only 4 claims with a sentiment lower than -0.6 and all of them are false claims. As opposed to that, we find only 5 claims with a sentiment above 0.6 and they are either true or false.

Generation of more data may pour more light into this observation. We have only just started observing patterns that relate the credibility of Internet Claims to their sentiments. This will help us understand the reason why people choose to spread fallacious claims. Our observations, based on limited data, suggest that highly negative claims which burgeon negativity among masses have a tendency of being fallacious. This tell us that false claims are made to bring about a sense of negativity among masses.

6 CONCLUSION

The research aimed at fake news analysis, which has data labeled with ratings of news/claims/tweets/internet articles, to perform certain machine learning predictions, needs to verify the credibility of those ratings before proceeding with the prediction. This paper formally defines the conditions necessary for assessing the credibility of the ratings of internet claims. This paper also describes a methodology used to create a data set for analysis of internet claims or fake news analysis and any predictive research aimed at the detection of fake news online. In the process of creating this data set, we also made observations about the sentiment behind these claims and compared the true positives with the false positives and true negatives with false negatives. We observed that both true and false claims had about 31% of the claims positive and the rest, negative. This means we cannot generally comment on the sentiment of a false or a true claim. But we can conclude that, if we have a highly negative claim, it has a tendency of being false. We also conclude after looking at the numbers that there were more false news articles/false claims than true news articles/true claims. In any given random sample, there is more fake news than reality.

7 FUTURE WORK

This research needs to be further expanded to use techniques for data collection as described here to extract an analyze more internet claims to create a large scale corpus of internet claims. We also need to use the claims from this data set separately for every social media platform, to analyze the time series data of such claims spread by people on those media. Only on proper analysis of such time series

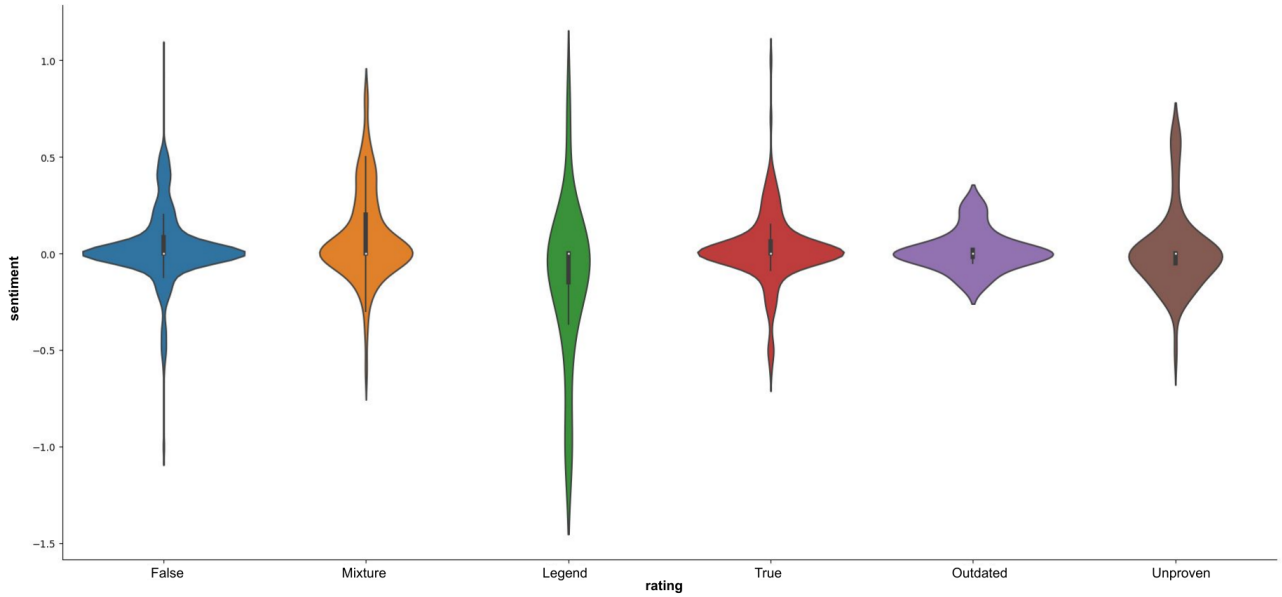

Figure 6: Violin Plot of Clustered Ratings

data can we come up with conclusive machine learning algorithms to predict if any given claim is true or false. Furthermore, research aimed at understanding the psychology of people is necessary to understand why fake news travels much faster than truth and why there is more fake news and false claims over the internet than truth.

REFERENCES

[1] [n. d.]. Data set to accompany Truth Discovery and Fact Checking: Theory and Practice SIGKDD 2019 Workshop, August 5th, Anchorage, Alaska paper 'A Data Set of Internet Claims and Comparison of their Sentiments with Credibility'. https://github.com/the-lost-explorer/iClaimNet.

[2] Scientific American. 2018. Biases Make People Vulnerable to Misinformation Spread by Social Media. https://www.scientificamerican.com/article/biases-make-people-vulnerable-to-misinformation-spread-by-social-media/.

[3] Crummy.com. [n. d.]. Beautiful Soup-a Python library for pulling data out of HTML and XML files. https://www.crummy.com/software/BeautifulSoup/bs4/doc/.

[4] Tanushree Mitra and Eric Gilbert. 2015. CREDBANK: A Large-scale Social Media Corpus With Associated Credibility Annotations. https://github.com/compsocial/CREDBANK-data.

[5] Indiana University Observatory on Social Media. [n. d.]. Visualize the spread of claims and fact checking. https://hoaxy.iuni.iu.edu/.

[6] Pandas.pydata.org. [n. d.]. Pandas - Python Data Analysis Library. https://pandas.pydata.org.

[7] Amey Parundekar. [n. d.]. Regular Expression for URL Extraction. https://regexr.com/496mu.

[8] Vincent Russo. [n. d.]. Vincent Russo's GitHub repository on tweepy API. https://github.com/vprusso/youtube_tutorials/blob/master/twitter_python/part_1_streaming_tweets/tweepy_streamer.py.

[9] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. [n. d.]. Fake News Detection on Social Media: A Data Mining Perspective. https://www.kdd.org/exploration_files/19-1-Article2.pdf.

[10] Snopes.com. [n. d.]. Comprehensive list of the ratings snopes.com uses and their definitions. https://www.snopes.com/fact-check-ratings/.

[11] Sinan Aral Soroush Vosoughi, Deb Roy. 2018. The spread of true and false news online. http://science.sciencemag.org/content/359/6380/1146/tab-pdf.

[12] TextBlob.com. [n. d.]. TextBlob: Simplified Text Processing. https://textblob.readthedocs.io/en/dev/.

[13] Tweepy.org. [n. d.]. An easy-to-use Python library for accessing the Twitter API. https://www.tweepy.org.

[14] Twitter. [n. d.]. Twitter API parameters. https://developer.twitter.com/en/docs/tweets/timelines/api-reference/get-statuses-user_timeline.

[15] Twitter. [n. d.]. Twitter's Platforms for developers. https://developer.twitter.com/en/docs/basics/getting-started.