Sentiment and Retweet Analysis of User Response for Early Fake News Detection

Uzoamaka Ezeakunne[1], Shuyuan Mary Ho[1][0000-0002-4790-1821], Xiuwen Liu[1][0000-0002-9320-3872]

[1] Florida State University, Tallahassee FL 32306, USA
ufe18@my.fsu.edu, {smho, xliu}@fsu.edu

Abstract. Social media users have the freedom to propagate information, but this freedom has come with a price as social media has also become a channel to disseminate false information. It is thus vitally important to detect fake news in social media based on limited data available during the early stage of information propagation. This study introduces a novel approach to fake news detection and explores features e.g., retweet rate and sentiments based on the intuitions that fake news spreads faster and attracts more sentiment respectively to classify news as fake or real. The study contributes to fake news early detection by analyzing the feature sets associated with news posts on social media using limited data. The paper highlights the challenges of relevant feature sets to improve fake news detection accuracy.

Keywords: fake news, disinformation, social media, deep learning.

1 Introduction

Rampant propagation of fake news has become a crucial problem, which impacts the stability of the society. An example of the consequence of fake news is the "pizzagate" incident [1] where fake news led to a real shooting. On the social media platform, it is difficult to distinguish and assess the authenticity and legitimacy of the news at a glimpse. Moreover, the adoption of social media increased the dissemination of fake news due to unregulated distribution of information in social media that has basically provided everyone a global platform to become an author of information—whether real or fake. This convenience of publishing information online has reduced the quality of information being produced—in comparison with the news and information propagated by legitimate, qualified news agencies.

Fake news can come in different forms (e.g., text, images, videos, etc.) in social media. Also, due to the magnitude of the consequences of fake news spread on social media, it is important to detect them early and stop their propagation automatically; this is a challenge due to insufficient data at the early stage of news propagation. Our research objective is to study the detection of fake news on-time for all-inclusive forms on social media. In order to achieve this, we limit the data used and study features available in all forms of news; specifically, if two news were posted on social media, one was an image and the other was a text, instead of using text as a feature, we use features (metadata) that are available to both forms (e.g. user response) and then set a limit to the amount of metadata used for each news.

Our study attempts to detect fake news in several forms by exploring features available during the early phase of news propagation. More specifically, we answer the following research question: *Can limited metadata obtained from a social media news post at its early stage detect fake news accurately*? To the best of our knowledge, we are the first attempt at this research question.

In the subsequent sections, we will review and discuss the state-of-the-art dataset and the related methods in fake news detection. Next, we discuss our research framework and results of the experiments. Finally, we highlight challenges faced in fake news detection as well as propose solutions for future study and conclude the paper.

2 Related Works

Currently, there are varying datasets as well as varying methods adopted by researchers to successfully perform automated detection in fake news and deception detection. In this section, we will briefly review and describe the related works.

2.1 Existing Datasets

The datasets used by researchers in deception detection can be grouped into three categories; crowd-sourced dataset, fact-checking websites, and derived datasets.

Crowdsourced Datasets. Researchers often perform deception detection with the use of crowdsourced dataset. This means that these datasets are results of a simulated environment. An example is the detection of negative opinion spam by Ott et al. [7] where Human Intelligence Tasks (HITs) were distributed evenly among hotels to write fake negative reviews. These people were instructed to assume that they work as marketers in a different hotel and the marketing manager asked them to write fake negative reviews against a competitive hotel. These crowdsourced reviews were used as dataset to aid deception detection by Ott et al. [7].

Fact-Checking Websites. In order to provide real-world data to support fake news detection, there are fact-checking repositories that are available to researchers. These fact-checkers are websites that contain several news that have been distributed in different domains, gathered and labeled by trained professionals who find relevant evidence to label statements. Examples of fact-checkers are PolitiFact[1] – fact-checker

[1] https://www.politifact.com

mainly for news on politics, and Gossip Cop[2] – fact-checker for news on celebrity gossips, to mention a few.

Derived Datasets. These derived datasets are mostly derived from fact-checking websites. Two derived datasets below namely LIAR dataset and FakeNewsNet are described.

LIAR Dataset. Wang et al [3] introduced 12.8k decade long, real world, manually labeled dataset for fake news detection. The dataset's news source is from the diverse PolitiFact[1] fact-checker, accumulated using its API. In order to verify the editor's reliability, Wang et Al (2016) went through a randomly sampled subset of the editor's analysis reports after which they concluded that 82% of the reports are reliable. LIAR dataset contains an ID number, the statement, the subject(s), the speaker, the speaker's job title, the state info, the party affiliation, the total credit history count (the total number of true, false, half-true, mostly-true and pants-on-fire counts in the dataset respectively) and finally, the context (venue/location of the statement).

FakeNewsNet. This is a data repository created to support fake news detection in social media [4]. This data repository of over 1000 news contains a link to the news content, the title of the news and the ids of social media posts related to that news on twitter. The news sources are from two fact-checking websites; gossipcop[2] and PolitiFact[1] and have been binary classified as either fake or real.

2.2 Existing Methods

Several methods have also been introduced to combat fake news. One of them is the introduction of fact checking websites such as PolitiFact[1], Gossip Cop[2], Snopes[3] etc. These websites contain previously shared news, which have been labeled by journalists as real or fake. These fact-checking websites also serve as repositories to support researchers in automatic fake news detection.

Several researches have been done towards detecting fake news automatically by using linguistic features from text [3] [4] [5]; based on the intuition that liars have peculiar linguistic patterns. The use of hybrid approaches was introduced by researchers which involves the use of text and other information such as user history [3], other users' responses [4] to mention a few. However, these methods are not all-inclusive for different news forms; they rely on text from news sources for detection.

Automated deception of fake news has been interesting for several researchers who have different approaches to solving the problem. These approaches involve the use of different feature set and different models; some of these feature set include; linguistic features of news text, meta data from news source users and network information. Also, the state-of-the-art methods combine these feature set and artificial intelligence models to improve accuracy in fake news detection.

[2] https://www.gossipcop.com
[3] https://www.snopes.com

The use of linguistic features to detect fake news was used as news writers tend to have a different linguistic pattern than writers of real news. In [3], [4] and [5] to mention a few, linguistic features were used to detect fake news. Wang et al [3] used linguistic features on short text description of news and Shu et al. [4] used it on longer news articles. Also, Su et al. [8] used linguistic pattern to compare 321k news titles in order to discover news titles that either disagree, agree or are unrelated with one another. Su et al. [8] also stated that in the relationship between news titles can detect fake news as news articles that disagree with one another, such a news pair is suspected to contain fake news.

The use of meta data of source users was used by Shu et al. [5] to computationally detect fake news with proper explanation on social media by exploiting user comments as well as news contents for fake news detection. Also, Wang et al. [3] used the truthfulness history vector of users, users' job, state and party as features for fake news detection. The truthfulness vector is simply the number of news for a particular user put in a vector according to the label of that news. For example, a user with news posts can the count of real news and count of false news in a vector and the sum of these count is the total count of news by a user.

Network information have also been explored [4] [5]. This involves the use of social behavior to detect fake news. Shu et al. [4] exploited social context information such as the sentiment in users' comments in response to a news as a part of the feature set of fake news detection. In [5], Shu et al also harnessed spatial temporal information for detection.

3 Research Framework

The objective of this study is to detect fake news accurately using the limited metadata available during the early stage of news propagation on social media. The research framework processes involve; (i) data collection and (ii) classifying the data based on the features collected in data collection stage. The features explored are retweet rate and sentiment of up to 100 posts based on the intuitions that the responses from fake news have more sentiment [4] and fake news attract more attention that real news [6]. In this section, we will briefly describe the processes involved in our research (see Fig. 1). In the next section, we will explain in depth how each process was achieved.

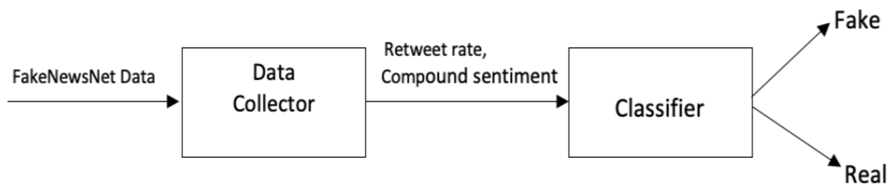


Fig. 1. Research framework

During data collection, we imported the FakeNewsNet dataset into a data collection algorithm (see Fig. 2) in order to calculate the retweet rate and compound sentiment of each news as our feature set. Next, every news along with its feature-set is fed into a classifier in order to classify the news as real or fake.

4 Experimentation

In this section, we will describe our research method in depth. Firstly, we will get familiar with some social media concepts and tools used. Next, we will show the implementation of each process in the framework in Fig. 1.

4.1 Social Media Concept

Twitter is a social network where users can post information which could be a *tweet* –posted originally by the user sharing it, or a *retweet* – re-posted a tweet by other users. Each of these posts have an *id*. In order to get data from this social network, we use the *Twitter API*.

4.2 Tools Used

Vader sentiment analysis tool. This is an open-source rule-based sentiment analysis tool suitable for analyzing social media text [9]. This takes in a text and returns a vector of the degree of positivity, a compound score, followed by degrees of neutrality and negativity in a range of zero to one. For example, a positive text should return a vector like this:

{'pos': 0.754, 'compound': 0.9227, 'neu': 0.246, 'neg': 0.0}

The sum of the positive, negative and neutral scores sum up to one. The compound score also called the composite score is derived from the other sentiment metrics and can be a negative value for a negative text, a positive value for a positive text and zero value for a neutral text.

Scikit-learn. This is a machine-learning library [10]; available in Python for integrating machine-learning algorithms for supervised and unsupervised problems.

4.3 Dataset

The dataset used is the FakeNewsNet dataset from which only up to 100 post ids (for posts with over 100 retweets) of twitter users who posted a particular news were collected. For each news, the average compound sentiment and average retweet rate were calculated (see Fig. 2). The average sentiment for a news was calculated using the vader sentiment tool to get the compound score from the text of other users who retweeted the news. Then, we got the average of all the compound sentiments in the retweets for a particular news post. On the other hand, the average retweet rate was collected by getting the average of all retweet-followers ratio. In order to get relevant data, we eliminated news posts with count of retweet of less than 3 and analyzed not more than 100 retweets because our research is based on limited data. In Table 1, the resulting data is a total of 621 news and 36,950 retweets analyzed. The input to this phase was the FakeNewsNet dataset which was read from file and loaded into a two-dimensional vector where each row corresponds to a news post, the output is the retweet rate and compound sentiment score for each 621 news (along with their PolitiFact ids and labels).

```
Algorithm 1: Data_collector
F is a 2D Vector
    C←∅
    R←∅
    for each row in F:
        y=0, z=0, count=0.
        for each column p in F:
            m = Get_Retweet_Rate(p)
            n=Get_Comp_Sentiment(p)
            y = y + m
            z = z + n
            count = count+1
            If count ==100:
                break
            end if
        end for
        r = y/count, c = z/count
        C←C+{c}, R←R+{r}
    end for
return C, R
```

Fig. 2. Data collection algorithm

Table 1. Size of data

	Fake	Real	Total
Number of news	316	305	621
Number of posts for all news	19,803	17,147	**36,950**

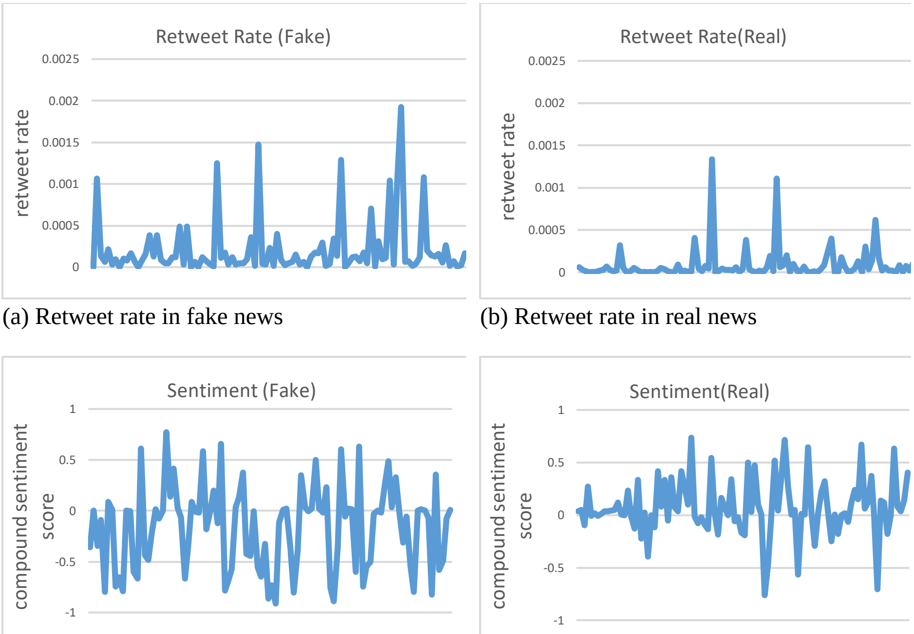
Next, the dataset was divided into the training set and test set in the ratio of 70:30 for number of news in the training set to number of news in the test set (Table 2). The training data was made up 434 news and 26,151 posts and the test set was made up 185 news and 10,799 posts.

Table 2. Size of training and test data

	Train	Test
Number of news	434	185
Number of posts for all news	26,151	10,799

4.4 Feature Analysis

A random sample of the retweet rate and compound sentiment score gathered during feature-set collection were visualized in Fig. 3. It can be observed that fake news has more retweet rate than real news (illustrated in Fig. 3 (a) and (b)). Also, the sentiment in fake news are less neutral than real news; real news sentiment is closer to the neutral sentiment value zero as seen in Fig. 3 (c) and (d).



(a) Retweet rate in fake news (b) Retweet rate in real news

(c) Sentiment in fake news (d) Sentiment in real news

Fig. 3. Visualized feature-set

4.5 Classification.

Each news in the test set was assigned to a label first by using random classification and then Support Vector Machine (SVM), logistic regression, Naïve Bayes and a Deep Neural Network (DNN) were implemented for performance comparisons. The implementation of the classifiers is briefly described below:

- *Random.* Random labels were assigned to each news.
- *SVM, Logistic Regression and Naïve Bayes.* These classifiers were implemented using the scikit-learn tool [10] with 20-fold cross-validation.
- *Neural Networks.* This was a deep neural network, made up of seven layers with batch size of 80 and 50 epochs.

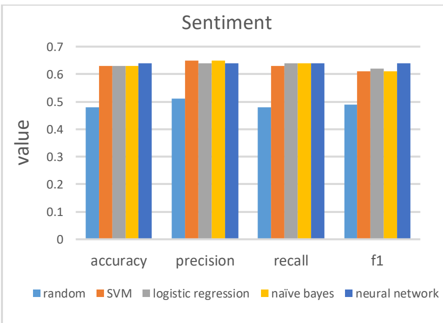
4.6 Performance Results

The performance metrics used in the experiments are accuracy, precision, recall and F1. After classifying the news gotten from the dataset, using the sentiment and retweet rate as features, classifications based on SVM, logistic regression, Naïve Bayes and neural network models were done as well as the random labels. From the results of the experiments in Table 1, we made some observations. First, when the retweet rate was used as the only feature, the performance of classification was not significantly better than using random labels. Next the use of sentiment alone performs as much or better than the use of both retweet rate and sentiment as features in the classifiers. However, the Naïve Bayes classifier records an improvement in precision when combining both features.

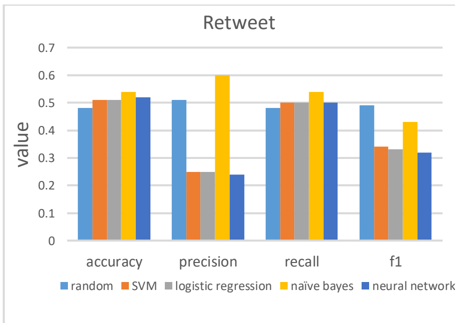
Table 3. Experimental results

	Accuracy	Precision	Recall	F1
Random	0.48	0.51	0.48	0.49
SVM (retweet)	0.51	0.25	0.50	0.34
SVM (sentiment)	0.63	0.65	0.63	0.61
SVM (retweet + sentiment)	0.63	0.65	0.63	0.61
Logistic Regression (retweet)	0.51	0.25	0.5	0.33
Logistic Regression (sentiment)	0.63	0.64	**0.64**	0.62
Logistic Regression (retweet +sentiment)	0.63	0.64	0.63	0.62
Naïve Bayes (retweet)	0.54	0.60	0.54	0.43
Naïve Bayes (sentiment)	0.63	0.65	**0.64**	0.61
Naïve Bayes (retweet + sentiment)	0.58	**0.67**	0.58	0.51
Neural Network (retweet)	0.52	0.24	0.50	0.32
Neural Network (sentiment)	**0.64**	0.64	**0.64**	**0.64**
Neural Network (retweet + sentiment)	**0.64**	0.64	**0.64**	**0.64**

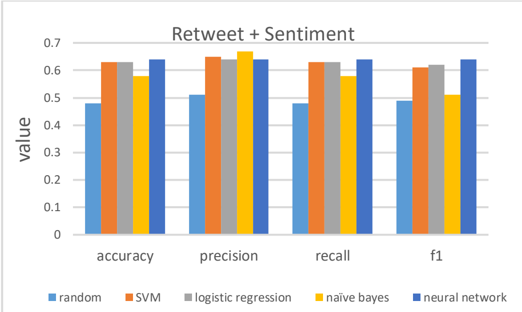
We observed that Naïve Bayes provides the best precision, while DNN offers the highest accuracy on the two features. Although the DNN, Regression, and Naïve Bayes all perform well with regards to recall value, also, DNN gives the highest F1 value. Also, it is seen in Fig. 4 that there is significant improvement in the use of sentiment as a feature with the neural networks outperforming other classifiers.



(a) Sentiment Feature Results



(b) Retweet Feature Results



(c) Sentiment + Retweet Feature Results

Fig. 4. Feature Results

To sum up, the study showcases that the sentiment of other users in response to a news is a relevant feature in early fake news detection when compared to retweet rate as features in early fake news detection.

5 Fake News Detection Challenges

Automatic fake news detection relies on abundant real-world data and relevant features. However, there are some challenges faced in solving this important problem of fake news detection such as the inability to merge feature sets, inconsistent API availability and the diminishing privileges available to researchers on social media API. These problems will be explained, and solutions will be proposed for each of them.

The first problem is the limitation in feature set exploration as a result of the inability to merge features from different datasets. Although some of the relevant datasets that support fake news detection are derived from the same fact-checker, it is still difficult to explore their various feature-sets due to the difference in the identifier used. In order to explore feature-sets, the identifiers (e.g. news id) used in these datasets should be from its source fact-checker to enable merging of relevant feature sets.

The second problem is the inconsistent API availability for Fact-Checking Websites. Due to the importance of the fake news detection problem, the availability of access to data to support research is equally important. Hence, fact-checking websites should consistently ensure the availability of their API.

Finally, due to privacy concerns on social media, the privileges on the APIs are declining. However, privacy is an important issue as well as fake news detection, hence, we propose that social media should provide APIs to research institutions and organizations with the privileges needed to support detection.

6 Conclusion and Future Work

Automated fake news detection is an important challenge in the society. However, it is crucial to detect fake news early in all forms. In this paper, we studied a novel research problem targeted at automatically detecting fake news early, in all forms, using limited data. We showed the performance of two features - retweet rate and sentiment of other users; obtained from these limited, real-world data on random, SVM, Logistic Regression, Naïve Bayes and DNN classifiers. The results indicate that the use of sentiment of other users in response to a news as a feature was more relevant when compared to retweet rate as a feature in early fake news detection with the DNN outperforming other classifiers. Finally, we discussed some challenges in fake news detection and proposed potential solutions to these problems. In the future, we intend to explore more features identified by researchers (e.g. user truthfulness history [3]) in order to discover other relevant features that work well with limited data in order to improve the accuracy of all-inclusive, early fake news detection.

References

1. M. Wendling: The saga of 'Pizzagate'. The fake story that shows how conspiracy theories spread, https://www.bbc.com/news/blogs-trending-38156985, last accessed 2020/06/25.
2. S. B. Parikh, P. K. Atrey: Media-rich fake news detection: a survey. In: IEEE Conference on Multimedia Information Processing and Retrieval (MIPR) 2018, pp. 436–441 (2018).
3. W. Y. Wang: "liar, liar pants on fire": A new benchmark dataset for fake news detection. In: Proceedings of the Association for Computational Linguistics Short Papers 2017, vol. 2. pp. 422–426. Association for Computational Linguistics (2017).
4. K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu: FakeNewsNet: A Data Repository with News Content, Social Context and Spatialtemporal Information for Studying Fake News on Social Media. In: The 9th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (2018).
5. K. Shu, L. Cui, S. Wang, D. Lee, and H. Liu: dEFEND: Explainable Fake News Detection. In: Proceedings of KDD, pp.395–405 (2019).
6. Z. Zhao, J. Zhao, Y. Sano, O. Levy, H. Takayasu, M. Takayasu, D. Li, and S. Havlin: Fake News Propagate Differently from Real News Even at Early Stages of Spreading. arXiv:1803.03443 (2018).
7. M. Ott, C. Cardie, and J. T. Hancock: Negative Deceptive Opinion Spam. In: Proceedings of NAACLHLT. pp. 497– 501 (2013).
8. T. Su, C. Macdonald, and I. Ounis: Ensembles of Recurrent Networks for Classifying the Relationship of Fake News Titles. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 893–896 (2019).
9. C. J. Hutto and E. Gilbert: VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. In: International AAAI Conference on Weblogs and Social Media. AAAI; 2014. pp. 216–225 (2014).
10. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, vol 12. pp. 2825–2830 (2011).