Industrial Engineering Journal
ISSN: 0970-2555
Volume : 51, Issue 10, October: 2022

FAKE NEWS DETECTION USING DATASCIENCE

M. Priyanka, Assistant professor, Computer Science Engineering, Anubose institute of Technology, Palvancha, Telangana, India.

D. Veeraswamy, Assistant Professor, Computer Science Engineering, Anubose institute of Technology, Palvancha, Telangana, India.

V. Kavitha, Assistant Professor, Computer Science Engineering, Anubose institute of Technology, Palvancha, Telangana, India.

ABSTRACT

The purpose of this thesis is to assist in automating the detection of Fake News by identifying which features are more useful for different classifiers. The effectiveness of different extracted features for Fake News detection are going to be examined. When classifying text with machine learning algorithms features have to be extracted from the articles for the classifiers to be trained on. In this thesis, several different features are extracted: word counts, n gram counts, term frequency-inverse document frequency, sentiment analysis, lemmatization, and named entity recognition to train the classifiers.

Two classifiers are used, a Random Forest classifier and a Naïve Bayes classifier. Training on different features combined with different machine learning algorithms yields different accuracies. By testing the different features on different classifiers, it can be determined which features are the best for Fake News detection Classifying news articles as either Fake News or as not Fake News is explored using three datasets, which in total contains over 40,000 articles. One of the datasets is used to partly to train the classifiers and partly to test the classifiers. The remaining two datasets are used purely for testing the classifiers.

Introduction

The purpose of this thesis is to assist in automating the detection of Fake News by identifying which features are more useful for different classifiers. The effectiveness of different extracted features for Fake News detection are going to be examined. When classifying text with machine learning algorithms features have to be extracted from the articles for the classifiers to be trained on. In this thesis, several different features are extracted: word counts, n gram counts, term

frequency-inverse document frequency, sentiment analysis, lemmatization, and named entity recognition.

Two classifiers are used, a Random Forest classifier and a Naïve Bayes classifier. Training on different features combined with different machine learning algorithms yields different accuracies. By testing the different features on different classifiers, it can be determined which features are the best for Fake News detection. Classifying news articles as either Fake News or as not Fake News is explored using three datasets, which in total contains over 40,000 articles. One of the datasets is used to partly to train the classifiers and partly to test the classifiers. The remaining two datasets are used purely for testing the classifiers.

OBJECTIVE

The goal of this research is to find the patterns that correlate with a piece of news which are potentially fake. Obviously, in any classification analysis there must to be human intervention at some time. Although, it may not be possible to achieve 100 percent accuracy, finding the commonalities of Fake News would be a step forward. For this purpose, the plan is to initially collect a large amount of data already known and verified as Fake News and try to train a model that will associate a piece of news with the probability of it being Fake News. To classify news articles the raw text data needs to be turned into something more useful. This is called feature extraction. Feature extraction can take many forms: word counts, n-gram counts, punctuation usage, sentiment analysis, and many others. The extracted features can then be used to classify the article that the features came from. Different features may give different results depending on the underlaying patterns in the data. By testing different classifiers with different features one can determine patterns in the data. By determining the best features for classifying Fake News the potential for automated Fake News detection can be increased.

FEATURES

To find patterns, several different features should be tested. Features are numeric values that describe the text. Examples of these numeric values are word count or the number of times a particular punctuation mark is used. Some features will be more helpful than others, for instance

the number of verbs is more likely to be useful compared to the number of times a particular word is used, such as 'kitten'. The goal is to find the features that are most helpful in detection of Fake News. Next, each extracted feature will be discussed in detail. Word counts are among the most easily obtained features that can be extracted from raw text. It is simply a count of all the terms in a body of text. Word counts are also called a 'bag of words', however, to keep names descriptive, we shall call this type of feature a count. To get the word count in texts, scikit-learn's CountVectorizer is used; the CountVectorizer tokenizes the data and then counts each term

. The data can be tokenized by word or by n-gram. N-grams are series of n items, such as words or characters. In this thesis n-grams refers to groupings of two and three characters. For instance, the n-grams of the word 'feature' would be as follows: 'fe', 'ea', 'at', 'tu', 'ur', 're', 'fea', 'eat', 'atu', 'tur', and 'ure'. These features will be referred to as countword and count-ngram respectively. Term frequency-inverse document frequency, or TF-IDF, is calculated as follows: term frequency times the inverse document frequency. Where term frequency is the number of times a term is in a document divided by the number of terms in a document. The inverse document frequency is the logarithm of the number of text (or articles) in the collection divided by the number of texts or articles where the term appears. Below is the equation for TF-IDF: TF-IDF = number of term occurrences terms in text × log number of texts in collection number of texts where term occurs

This is important for algorithms as they do not process the meaning of words. By labeling words as 'person' or 'organizations' algorithms can pick on patterns involving these entities that would otherwise be obstructed. For this thesis, spaCy's named entity recognition was used. This feature will be referred to as ER

RESULTS

Using two different models, each extracted feature was tested. The models used were Random Forest (RF) and Naïve Bayes (NB). There is some difference between the two classifiers. There is a much larger difference between datasets. The following is a detailed discussion of each set of features. We will compare features and classifiers by their accuracy, which is the percentage of correct classification made by the classifier

: Count Accuracies Count-word and Count-ngram: First, most notable the ISOT testing data is getting way higher accuracy results than either the Original dataset or the FakeNewsNet dataset. After the ISOT, the Original dataset is getting the next highest accuracy rates. This suggests that the Original dataset is closer in makeup to the ISOT dataset than the FakeNewsNet is. Next the data shows that the NB classifier generalizes better than the RF classifier.

The NB classifier gets better accuracy rates with count-ngram. The RF has no clear winner between count-word and count-ngram. RF Count-word RF Count-ngram NB Count-word NB Count-ngram ISOT News 97.42% 97.60% 94.74% 97.91% FakeNewsNet 54.98% 55.92% 57.11% 57.82% Original Data 74.12% 66.47% 75.88% 77.65% 40% 50% 60% 70% 80% 90% 100% Accuracy Count 17 : TFIDF Accuracies TFIDF-word and TFIDF-ngram: As seen in Figure 3, the ISOT testing data has the highest accuracies again. The random forest classifiers get better results with the ISOT dataset than the Naive Bayes.

However, the NB does generalize better to the Original dataset and the FakeNewsNet dataset. TFIDF-word is getting better accuracy rates over TFIDF-ngram. In the case of the RF's classification of the Original dataset, the TFIDF-word is getting 6.47% more accuracy. Again, the Original dataset is being classified better than the FakeNewsNet dataset. Between TFIDF and Count, the Count-ngram is getting the best accuracy results

Lemma: Once again, ISOT accuracies are the highest, with Original coming in second. The NB classifier is still generalizing better than the RF classifier. The results from lemma are better than some of the other features. However, lemma with a Count-word is not as accurate as a Countword. Suggesting that the different forms of a word are helpful to the classifier.

From the results a few more general conclusions can be made. The most notable is that the accuracy on the ISOT test data is much higher than the accuracies of the other datasets. From this, it can be concluded that there is a pattern in the ISOT dataset that is being picked up by the two classifiers. However, it appears that these patterns do not generalize well to the other available datasets. The patterns that the classifiers are picking up on could be a pattern found in Reuters articles, or could be another pattern that exists mainly in the ISOT dataset Such as article topic, or political leaning

Conclusions

With the nature of the Internet as it is, Fake News is easily created and distributed. Fact checking is tedious and time consuming, so automating Fake News detection is critical. Thus Fake News classifiers should be created. However, a classifier does not come out of thin air, it must be trained on already existing data. The quality and quantity of the data is important. Three datasets were used for the research in this thesis. ISOT, a huge dataset of over 40,000 articles. FakeNewsNet is another, much smaller dataset containing 422 articles. Lastly, the Original dataset, containing 180 articles, that was gathered specifically for this research. However, a classifier cannot read, so it must have features extracted for the articles. A feature is a numeric value extracted from the article. Such as a word count, or a count of parts of speech, or more complicated features

FUTURE ENHANCEMENT

Work A different dataset should be used to train classifiers to verify the result obtained with ISOT. The size of ISOT makes it a valuable dataset, however, it is probably best as a testing dataset than a training dataset. Using combinations of the features should be explored. For instance, combining ER with lemma. Even more testing with VADER scores could be beneficial.

The best accuracy rate on the Original data set was achieved with a Naïve Bayes classifier with a word count feature after NLTK stop words were removed. These results should be explored more, for example which words when removed provide the greatest increase in accuracy. Additionally, it should be look into if the removal of any the NLTK stop words actually harm the overall accuracy of the classification. One thing that has not been tried is differentiating and classifying real news, satire, and Fake News. This would be valuable because satire is a type of deceptive news that isn't Fake News. Hence, we should avoid labeling it as such.

REFERENCES

I. Witten, E. Frank, and M. Hall, Data Mining 4th Edition. 2016. [4]

2.W. Y. Wang, "'Liar, Liar Pants on Fire': A New Benchmark Dataset for Fake News Detection," in Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2017, pp. 422–426. [5]

3.D. Byrd, "The science of fake news gets a boost," 2018. [Online]. Available: http://earthsky.org/human-world/fake-news-mar-2018-article-science-calling-for-studies.