Mining for Credibility

Mahdokht Afravi
Jonathan Avila
mmafravi@gmail.com
jeavila6@miners.utep.edu
University of Texas at El Paso
El Paso, Texas

Cristian Ayub
Gerardo Cervantes
cayub@miners.utep.edu
gcervantes8@miners.utep.edu
University of Texas at El Paso
El Paso, Texas

ABSTRACT

We used the publicly available FakeNewsCorpus dataset that consists of articles labelled as one of 11 types. We used articles labelled as ‘fake’, ‘reliable’, and ‘conspiracy’ to extract features. We convert the data to numerical data by using a spare matrix representation and using term frequency. With this feature set, we apply clustering algorithms to the dataset, and are able to find useful information about about the data. We train a model using linear regression that would predict whether an article is ‘fake’ or ‘reliable’ and find modest results. Our project site is a GitHub web page that can be found at https://mahdafr.github.io/19s_cs5362-dm/ which contains the source code, a PDF copy of the presentation slides, and a PDF copy of this report.

CCS CONCEPTS

• **Computing methodologies** → *Information extraction.*

KEYWORDS

data mining, natural language processing, datasets, fake news, DBSCAN, kmeans, linear regression, articles, information retrieval, machine learning

ACM Reference Format:
Mahdokht Afravi, Jonathan Avila, Cristian Ayub, and Gerardo Cervantes. 2018. Mining for Credibility. In *Woodstock ’18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, Article 111, 3 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Fake news has been a constant concern for as long as news became widespread with the invention of the press in 1439 [4]. As there is not a concrete definition of what fake news means, we will use the dataset named FakeNewsCorpus [6]. This dataset labels many articles with a label into 11 different types, this corpus contains an adequate amount of data to use for unsupervised and supervised learning methods. This data set will allow us to demonstrate that fake and reliable news can be identified despite the meaning people may give to such a concept.

2 THE CORPUS

FakeNewsCorpus is a publicly available dataset containing over 9.4 million articles from 745 domains [6]. Each article is categorized as one of 11 types, including *Satire*, *Extreme Bias*, *State News*, *Junk Science*, *Hate News*, *Clickbait*, and *Political*. Four of the subsets are the focus of our project:

- ***Fake News***, which contains 928,083 articles. This subset is built from sources that create or distribute false information, or distort factual news reports.
- ***Conspiracy Theory***, which contains 905,981 articles. This subset is built from sources that are “well-known promoters of kooky conspiracy theories”.
- ***Proceed with Caution***, which contains 319,830 articles from domains that may or may not be factual.
- ***Credible***, which contains 1,920,139 articles from domains that follow traditional, ethical journalism practices.

Each article contains 15 fields, including *author*, *domain*, *title*, *type*, and *keywords*, which will prove useful to the project’s focus.

3 THE PROBLEM

There is a lot of data that will be used for these experiments. One of the problems is going through the data and making sure that it is adequate for use in this project. The data is noisy, which means it might not be suitable to use for the problem and may require further automated cleaning. We may improve our results by pre-processing the data. A few pre-processing techniques when working with text are: *stopword removal*, *lemmatization*, and *stemming*.

Our main goal is to see if we can train a model that, given the contents or body of an article, determine whether it is ‘fake’ or ‘reliable’ using the words as its features.

4 THE APPROACH

There are many available methods for analyzing large datasets. However, the most suitable method for the problem at hand is the DBSCAN (*density -based spatial clustering of applications with noise*) algorithm [1]. This algorithm has a few key advantages:

- does not require a specific number of clusters,
- finds shaped clusters,
- has notion of noise, and
- is robust to *outliers*.

After doing a preliminary analysis of the data, we will then work towards creating a model that can predict whether an article is credible or not. We will also evaluate other subsets of data: *Conspiracy Theory*, *Proceed with Caution*, and *Credible*. We will manually

Woodstock ’18, June 03–05, 2018, Woodstock, NY
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-9999-9/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

annotate a small sample of this data with whether it is credible or not and then evaluate over it.

5 THE PLAN

The project was run in three phases. The first phase consists of data cleanup and organization, where we will remove the unused subsets from the original dataset. In the second phase, we identified relevant features and perform feature analysis in the subset of data. We also analyzed the parameters, such as *eps*, or max radius for neighborhood, and *minPts*, or min points in a neighborhood, of the data for the DBSCAN algorithm. In the third phase, we designed a model to predict the credibility of an article.

6 PHASE I: DATA ANALYSIS

6.1 Data Clean Up

We decided to work with a subset of the original dataset, as this would make future processing more efficient and we believed we would receive very similar results and patterns. We begin by filtering the dataset; creating subsets for the types of articles we are interested in. Entries with incomplete fields are discarded.

6.2 Pre-processing

The FakeNewsCorpus dataset needs to be in a numerical representation since it is in text format. Therefore, we implemented the following techniques: stop word removal, stemming, lowercasing, tokenization, and punctuation removal. These techniques are used to reduce the dimension of the given dataset so that is easier to analyze later [5]. For instance, stop work removal consists of identifying and eliminating meaningless words within the dataset, such as articles, prepositions, and conjunctions. However, for this specific technique, we need a stop-list, which is a dictionary of stop words. Punctuation removal removed the punctuation from the dictionary, this was done because there is little information to gather from punctuation for this task. Lowercasing was done because uppercase of a word provides very little information for the algorithm to use and it would help make the word dictionary used have more unique words and be less sparse.

We created a dictionary that contained the most common words in the articles, we gave each word an index. For every article we created a vector, for every word in the article we put the term frequency in the index for the word. We did this for all the articles, and afterwards converted the matrix into a sparse-matrix representation. Due to the huge memory size of the matrix, this significantly reduced the memory consumption of the matrix.

7 PHASE II: FEATURE ANALYSIS

7.1 K-means

We used k-means clustering [7] to separate the dataset into subsets for further analysis. The labels used for the subsets were: conspiracy, fake, and reliable. The given dataset had $30,000$ articles. A dictionary of $20,000$ words was used to classify the articles properly in a different cluster. Table 1 shows the clusters obtained from k-means and how the data is distributed in them.

Table 1 shows how the $30,000$ articles were distributed among the eight clusters created. This clustering technique allowed us to analyze the given dataset by identifying how many articles are indeed reliable and belong to their corresponding label. For this technique we obtained an *average silhouette coefficient* of 0.217, meaning it is accurate but not as much as we would like.

Articles	Conspiracy	Fake	Reliable
19671	5499	6209	7885
7174	3168	2981	1025
1845	845	411	589
833	367	299	167
314	83	78	153
181	18	1	162
42	14	14	14
18	6	7	5

Table 1: Eight clusters, with number of articles for each label

We analyzed the labels of the articles in Table 1. We found that some clusters had many articles with centered around a specific label. For this reason, we looked at the cluster centers to find which words they were closest to. We found the cluster with 181 articles that had the majority of reliable labels to have the most important features be words from Spanish. Interestingly enough, there was mostly Spanish stop words since we only removed stop words from the English language. It is likely that reliable articles contains noisy data and is the only data source that data from Spanish which is why a majority were labeled as reliable, which means removing them would be beneficial for future processing.

7.2 DBSCAN

We used DBSCAN [1] to analyze how accurate the clusters obtained with k-means were. The average silhouette coefficient was poor and at times data might have different shapes and have too much noise. Table 2 shows the results obtained by implementing DBSCAN.

Matrix	Clusters	Noise
$15kwords \times 20k$ features	2	4.6%
$30kwords \times 20k$ features	3	4.3%

Table 2: Clusters created with a given dataset (articles × features)

We were able to understand that a higher *eps* allows us to generate better results. We were able to tune the parameters of DBSCAN to decrease the amount of noise and to correctly measure the distance from noise points to the nearest clusters' core points. We realized that both homogeneity and silhouette coefficient were bad metrics to use with DBSCAN.

However, this algorithm showed that much of the noise were articles from the *proceed with caution* subset as these articles required manual verification. Since k-means was able to give us more structured clusters, we trimmed our original subset to use only the features from the clustering of *credible*, *fake*, and *conspiracy* subsets to proceed with creating our model.

8 PHASE III: MODEL DESIGN

In this phase, the objective is a binary output on whether an article is credible or not. The features used for creating our machine learning model were our vocabulary with term frequency. The training data used comprised of $10k$ *reliable* articles and $10k$ *fake* articles. We used 75% of the articles as training data and the remaining articles as testing data.

8.1 Linear Regression

Linear regression was used as a machine learning model in order to classify a given article as reliable or fake. Linear regression model was based on the statistical model

$$y = \sum_{j=1}^{k} \beta_j x_j + \epsilon$$

where y is the dependent variable, x_j corresponds to the jth predictor, β_j is the jth regression coefficient and ϵ is a random error [3]. The implementation of this model over the defined dataset we obtained an accuracy of 66.3%. A threshold of 0.5 was used to convert the output of the linear regression model to a binary output {0,1}. The baseline of the model was to predict randomly, which produced a baseline accuracy of 50.0%. Thus our model did better than baseline and we were able to gain informative knowledge from whether an article is credible or not from the words used in the article. We first trained a model using linear regression, and then using naïve-bayes using the features extracted from Phase II.

8.2 Naïve-Bayes

Naïve Bayesian Classification (AKA Simple Bayesian Classifier) is the most known and used classification method [2]. This algorithm allows us to identify what is the probability that a given element X can belong to a specified class C. The basic relation formulated for this algorithm is

$$P(H|X) = \frac{P(X|H) \cdot P(H)}{P(X)}$$

Now, let us suppose we have our training set defined as a set of m samples $S = \{S_1, S_2, \ldots, S_n\}$, where the sample S_i is an n-dimensional feature vector $\{X_1, X_2, \ldots, X_n\}$. Additionally, there are k classes $s_1, s_2, \ldots, s_n$ and every sample belongs to one of such classes. Given an unknown data sample T we can predict the class it belongs to by using the highest conditional probability $P(C_i|X)$, where $i = 1, 2, \ldots, k$. Therefore, the basic idea of Naïve-Bayes Algorithm is formulated as follows

$$P(C_i|X) = \frac{P(X|C_i) \cdot P(C_i)}{P(X)}$$

Running the Naïve-Bayes algorithm, we were able to achieve a test accuracy of 92.2%. This was much better than the linear regression results, and shows that this algorithm was able to distinguish between credible and non-credible articles more accurately.

9 RESULTS AND DISCUSSION

We find that our models were able to predict whether an article is credible with an accuracy of 66.3% and an accuracy of 92.2% for our linear regression and Naïve-Bayes algorithms, respectively. Both algorithms achieved modest improvements over the baseline of predicting randomly. With the increasing number of articles spreading misinformation, a perfect model may not exist. However, this model can be used by the public to become more aware of deceptive sources. Our model shows term frequency, which is very important in distinguishing whether an article is credible or non-credible.

Contributions. Table 3 contains those aspects of the project to which each member contributed and/or maintained.

Member	Report	Analysis	K-means	DBSCAN	Models
Afravi	15%	25%	5%	85%	10%
Avila	15%	25%	5%	5%	40%
Ayub	45%	10%	15%	5%	5%
Cervantes	25%	40%	75%	5%	45%

Table 3: Group Members' Contributions to Group Project

REFERENCES

[1] Derya Birant and Alp Kut. 2007. ST-DBSCAN: An algorithm for clustering spatial–temporal data. *Data & Knowledge Engineering* 60, 1 (2007), 208–221.

[2] Cagatay Catal, Ugur Sevim, and Banu Diri. 2011. Practical development of an Eclipse-based software fault prediction tool using Naive Bayes algorithm. *Expert Systems with Applications* 38, 3 (2011), 2347–2353.

[3] Toby J Mitchell and John J Beauchamp. 1988. Bayesian variable selection in linear regression. *J. Amer. Statist. Assoc.* 83, 404 (1988), 1023–1032.

[4] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter* 19, 1 (2017), 22–36.

[5] C. Silva and B. Ribeiro. 2003. The importance of stop word removal on recall values in text categorization. In *Proceedings of the International Joint Conference on Neural Networks, 2003.*, Vol. 3. 1661–1666 vol.3. https://doi.org/10.1109/IJCNN.2003.1223656

[6] Maciej Szpakowski. 2018. FakeNewsCorpus. https://github.com/several27/FakeNewsCorpus. (2018).

[7] Kiri Wagstaff, Claire Cardie, Seth Rogers, Stefan Schrödl, et al. 2001. Constrained k-means clustering with background knowledge. In *Icml*, Vol. 1. 577–584.