Credibility Assessment of News on Social Media

JIANKUN WANG (A0231869U), LIUYANG CHEN (A0231850M), NANHAI ZHONG (A0231953E), YISHUN LIU (A0231849X), and YUANKAI MA (A0231868W), Business Analytics Centre, National University of Singapore, Singapore 119613, Singapore

1 INTRODUCTION

Nowadays, because of information explosion, information screening becomes much more difficult. News screening is one of the most important, since the problem of people believing more in fake news than true news may cause threats to society. In this project, we will leverage upon text analytics techniques, machine learning tools and self-learning strategy to establish an automated credibility assessment model for social media news. In addition, as we all know that texts on a social media platform tends to share similarity. In order to eliminate the platform-induced bias, we introduce three datasets from different sources and use self-learning.

2 DATASETS

2.1 Overview

We combine four fake and real news datasets in order to have enough data for the text news classification model and eliminate platform-induced bias. Two of them are labeled (Dataset I and II), which will be used to build the preliminary model. The third dataset (Dataset III) is unlabeled but has interactive information features, which will be used to conduct self-training and interactive information exploratory.

All of the three datasets are retrieved from Kaggle. Dataset I [1] was used by Ahmed H in his papers. Dataset II [2] was assembled by the creator from a competition hosted on dockship.io. Dataset III [3] was sourced from a chapter of the Machine learning for Cyber Security cookbook.

Table 1. Dataset Description

Dataset	Records	String	Datetime	Integer	Remark
I	12999	title, text, subject	date	/	False news and real news are in separate files.
II	44898	title, text, subject, class	date	index	/
III	44000	uuid, author, title, text, ord_in_thread, language, site_url, threa_title, country, main_img_url	published, crawled	main_rank, likes, spam_score, shares, comments, type, replies_count, participants_count,	Spam_score means the possibility that the news is fake.

2.2 Exploratory Data Analysis

In this part, we examine our data from various aspects. First, the balance of our labeled data is checked. As shown in this plot, no imbalance problem is spotted.

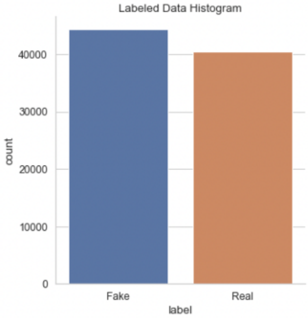

Fig. 1. Labeled Data Histogram

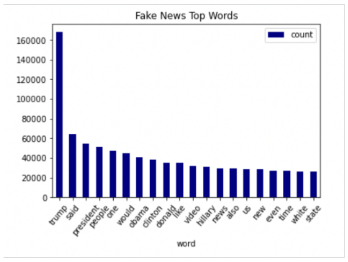

Fig. 2. Fake News Top Words

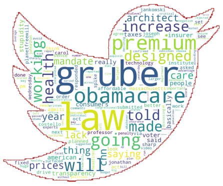

Fig. 3. Word Cloud

2.3 Data Preprocessing

Before preprocessing, construct the labeled datasets and the unlabeled datasets separately. For labeled datasets, remove the records with missing data and keep only three features: id, label and text. The text feature is a combination of both the original title feature and context feature. If there is no unique key in the original dataset, generate a unique key as id. Otherwise, use the original index feature or uuid feature as the id feature. For the unlabeled dataset, remove the records with missing data and has different language than English. Also, combine all the textual features together as a new text feature.

Then, using Map Function in PySpark to preprocess the text feature: Convert all the text feature to lowercase. Remove noise such as numbers, punctuations and stopwords. Tokenize and lemmatize the words by defining part of speech tag and use WordNetLemmatizer to transform all the words into dictionary form.

After preprocessing, vectorize the text features into Count Vector Matrix and TF-IDF Vector Matrix using the packages in PySpark and Sklearn. The Count Vector Matrix is a matrix with terms as the rows and document names as the columns and a count of words as the cells of the matrix, while the TF-IDF Vector Matrix is a similar matrix but with the TF-IDF as the cells.

3 PRELIMINARY MODELING

3.1 Pipeline

Now we have two preprocessing pipeline of text data. But which one of them should be chosen is determined by the downstream task. Here we will utilize three machine learning models to assess the credibility of news and evaluate the performance of the models. Each preprocessing methods together with the models will be evaluated and we will have a winner pipeline for later self-learning stage.

For each of the vectorized text dataset, we will follow the following stages: 1) Spilt the dataset into training set and test set. (70% training, 30% test) 2) Use the training set to build 3 machine learning models: Decision Tree, Random Forest, XGBoost. X is vectorized text, Y is label of the news. 3) For each model, Grid Search the best hyperparameters combination by a 5-fold Cross Validation. 4) Predict the label of the test dataset with the best models from step 3. 5) Evaluated the model accuracy in the test set with the true label and the predict label.

3.2 Model Performance

The performance of all the models is shown below. We choose the best pipeline based on the test set accuracy.

Table 2. Model Performance

Vectorization	Machine Learning Model	Hyper Parameters	Test Set Accuracy
Count Vectorized	Decision Tree	max_depth = 1000	84.40%

	Random Forest	max_depth = 15
		n_estimators = 2000
			73.50%

	XGBoost(GBDT)	max_depth = 6
		n_estimators = 1000
		learning_rate = 0.3
			85.00%

TF-IDF Vectorized	Decision Tree	max_depth = 50000	82.10%

	Random Forest	max_depth = 15
		n_estimators = 500
			73.70%

	XGBoost(GBDT)	max_depth = 6
		n_estimators = 1000
		learning_rate = 0.3
			83.40%

Our baseline accuracy is 52.20% if we just predict all the news to be fake. All our models can outperform the baseline model. Since the XGBoost with the Count Vectorized dataset can give us the highest test accuracy, we decide to move forward with the Count Vectorzied » XGBoost pipeline.

4 SELF-TRAINING

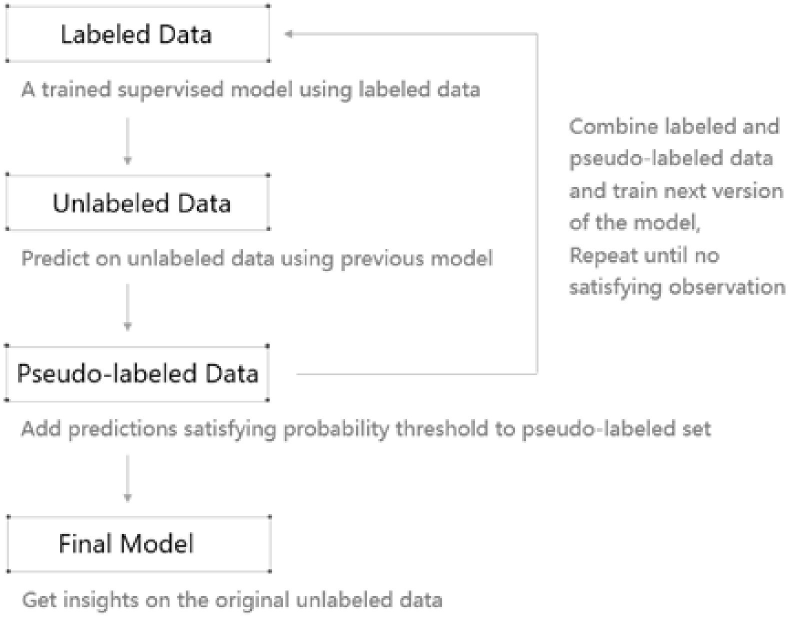
Labeled Data
A trained supervised model using labeled data
Unlabeled Data
Predict on unlabeled data using previous model
Pseudo-labeled Data
Add predictions satisfying probability threshold to pseudo-labeled set
Final Model
Get insights on the original unlabeled data
Combine labeled and pseudo-labeled data and train next version of the model,
Repeat until no satisfying observation

Fig. 4. Self-Training Pipeline

4.1 Purpose

Since our purpose is to analyze user behavior on the unlabeled dataset, we need to find a proper way to label this dataset. Our supervised models are trained on the combination of two other datasets, which is very likely to cause the model to overfit on the training datasets and comprise the performance on the unlabeled dataset. Therefore, we decide to use the self-training method, which is widely used in semi-supervised learning area, to help reduce variance of our models.

4.2 Pipeline

The pipeline of our self-training process is presented in Fig. 4.

4.3 Results Comparison

Compared with the preliminary model, the probabilities are more polarized because of the setting of threshold. Compared with the spam scores, the results after self-train are more balanced.

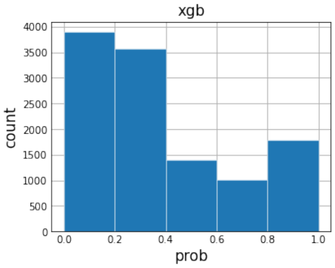

Fig. 5. XGBoost Model

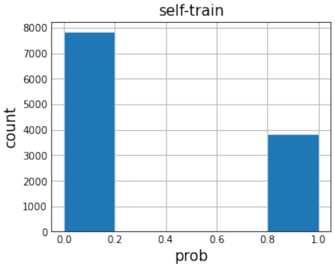

Fig. 6. Self-Training Model

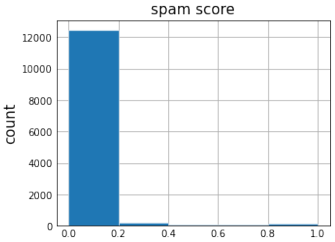

Fig. 7. Spam Score

5 INSIGHTS

5.1 Top 5 Authors of Fake and True News

Based on the total amount of news they published, we identified the top 5 authors of real and fake news.

Accordingly, we can judge the two authors ('EdJenner' and 'noreply@blogger.com (Alexander Light)'), who only appear in the fake news section above, with a high probability of publishing fake news.

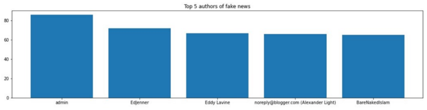

Fig. 8. Top 5 Authors

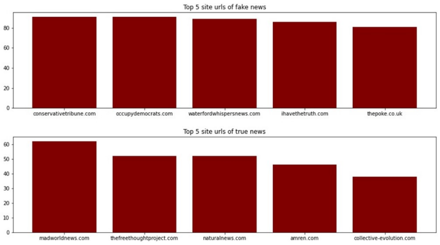

Fig. 9. Top 5 Sites

5.2 Top 5 Site URLs of Fake and True News

We also identified the top 5 Site URLs of publishing real and fake news.

Since there's no overlap, it is reasonable to classify the five sites ('conservativetribune.com', 'occupydemocrats.com', 'waterfordwhispersnews.com', 'ihavethetruth.com', 'thepoke.co.uk'), which each posted more than 80 pieces of fake news, as fake news sites.

5.3 Proportion of News Release Countries

The distribution of news release countries shown below indicated that there was little difference in the proportion of fake news and true news release countries, with the United States and Great Britain dominating both categories. This is mainly due to the fact that most news data sources come from these two countries.

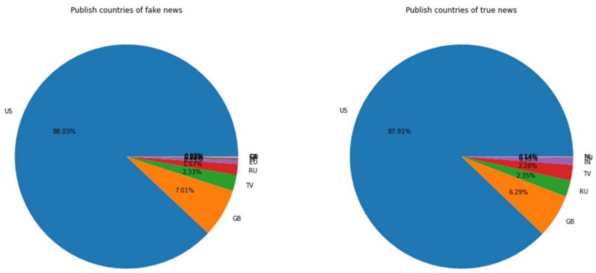

Fig. 10. Publish Countries

5.4 Word Cloud of Fake and True News Contents and Titles

Through the word clouds of fake news and real news' contents & titles, we can find out that the high-frequency words in the two categories are roughly similar. This actually indicates that, it is actually pretty difficult to judge one piece of news as fake or true only based on those single words that appear frequently in its title or content.

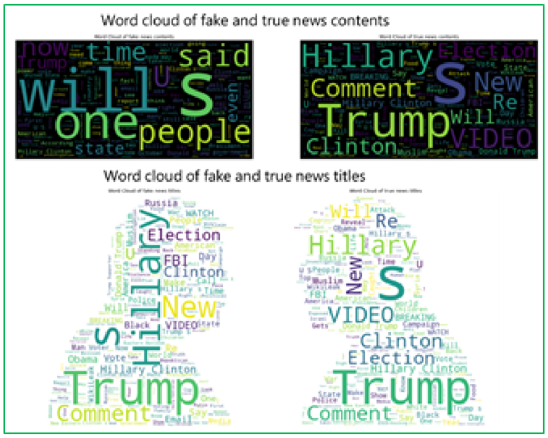

Fig. 11. Word Cloud of Fake and True News

5.5 Participants, Comments, Shares and Likes of Fake and True News

When looking at the participants, comments, as well as the shares and likes statistics, it's obvious that there exist significant differences between fake news and real news in these features. It is clear that fake news, compared to real news, would attract much more participants and get more comments, likes and shares. These characteristics could also provide important reference and reminders for us to screen fake news.

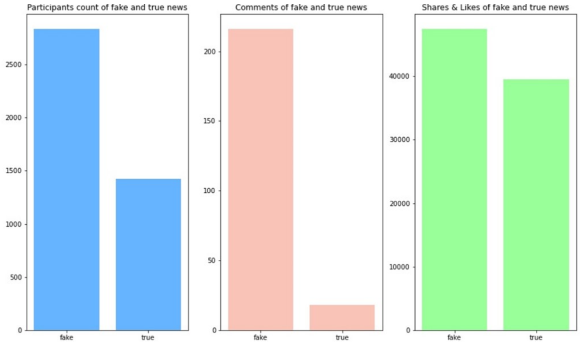

Fig. 12. Participants, Comments, Shares and Likes of Fake and True News

6 CONCLUSION

In this project, firstly, by extracting information from news headlines and body content from the first two labeled datasets, a text news classification model is built as the preliminary model to detect the textual similarity in different categories. Then, we conduct self-learning based on the preliminary model to predict on an unlabeled dataset with interactive information and compare the predicting results with spam scores. Finally, we take the interactive information features into account, such as the number of replies, the number of participants, etc., to find the relationship between them and the credibility of news.

This methodology is useful when there is only limited amount of labeled data. In practice, if the equipment conditions can support, pre-training the preliminary model on a larger dataset with records from various sources will be recommended. Also, when the model applied to a dataset on a specific platform without abundant labeled data, we could manually label several records and then use these records to conduct self-train. Therefore, the self-training process will fine-tune the model to better predict this dataset.

REFERENCES

[1] Hadeer Ahmed, Issa Traore, and Sherif Saad. 2017. Detection of online fake news using n-gram analysis and machine learning techniques. In *International conference on intelligent, secure, and dependable systems in distributed and cloud environments*. Springer, 127–138.

[2] Pankaj Gupta. [n.d.]. Fake News Dataset. [EB/OL]. https://www.kaggle.com/datasets/pnkjgpt/fake-news-dataset?select=train.csv Published 26 Oct. 2020.

[3] Emmanuel Tsukerman. 2019. *Machine Learning for Cybersecurity Cookbook: Over 80 recipes on how to implement machine learning algorithms for building security systems using Python.* Packt Publishing Ltd.