Fake News Detection Using Machine Learning Ensembled Methods.

Università della Calabria, Dipartimento di ingegneria informatica, modellistica, elettronica e sistemistica (DIMES), Rende (CS), Italy

Santiago Tibanquiza, Rocío Diaz, Mohammad Eqbal Balaghi

Abstract — This paper aims to detect the fake news using Machine Learning Ensembled Methods through the analysis of data. In this work, we propose to use machine learning ensemble approach for automated classification of news articles. Our study explores different textual properties that can be used to distinguish fake contents from real. we train a combination of different machine learning algorithms using various ensemble methods and evaluate their performance on 1 real database.

1. INTRODUCTION

Nowadays, the advent of the World Wide Web and the rapid adoption of social media platforms (such as Facebook and Twitter) paved the way for information dissemination that has never been witnessed in the human history before.

There has been a rapid increase in the spread of fake news in the last decade. Such proliferation of sharing articles online that do not conform to facts has led to many problems not just limited to politics but covering various other domains such as sports, health, and also science.

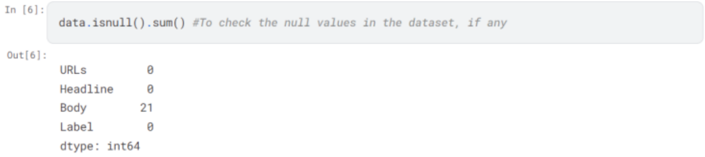
A few studies have primarily focused on detection and classification of fake news, the knowledge is then expanded to generalize machine learning (ML) models including support vector machine (SVM), logistic regression (LR), linear support vector machine (LSVM), decision tree (DT), and stochastic gradient descent (SGD) and the ensembled methods.

2. DATASET PRE-PROCESSING

The dataset required for the analysis was provided by Kaggle which contains a total of 4008 articles used for testing. The dataset is built from multiple sources on the Internet. The articles are not limited to a single domain such as politics as they include both fake and true articles from various other domains.

Data cleaning

the first step was to check how many null values exist in the dataset.



```
In [6]:  data.isnull().sum() #To check the null values in the dataset, if any

Out[6]:
         URLs         0
         Headline     0
         Body        21
         Label        0
         dtype: int64
```

Figure 1: Check the null values.

second, we must remove the null values.

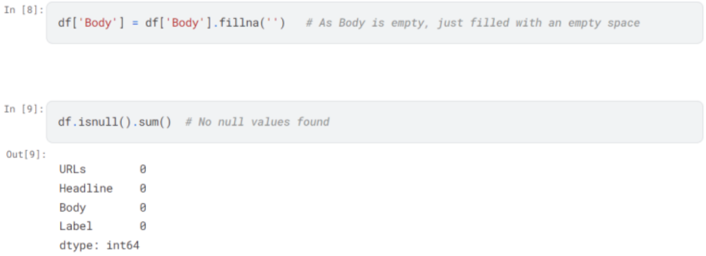
As Body field has some empty fields, it can be handled in two ways:

Drop the 21 rows.

Replace the null value with a dummy string.

Here, we will be going with the 2nd option, because although dropping 21 rows would not affect the accuracy, as it is just a minute portion of our large dataset, it is never recommended.

we will be replacing the Null(Nan) values in 'Body' field with an empty string ('')



```
In [8]:  df['Body'] = df['Body'].fillna('')   # As Body is empty, just filled with an empty space


In [9]:  df.isnull().sum()  # No null values found

Out[9]:
         URLs        0
         Headline    0
         Body        0
         Label       0
         dtype: int64
```

Figure 2: Replace the null values with empty string.

Once the relevant attributes are selected after the data cleaning and exploration phase, the next step involves extraction of the linguistic features.

These features include percentage of words implying positive or negative emotions; percentage of stop words; punctuation; function words; informal language; and percentage of certain grammar used in sentences such as adjectives, preposition, and verbs.

3. ALGORITMS

We used the following learning algorithms.

Logistic Regression.

As we are classifying text based on a wide feature set, with a binary output (true/false or true article/fake article), a logistic regression (LR) model is used, since it provides the intuitive equation to classify problems into binary or multiple classes.

Mathematically, the logistic regression hypothesis function can be defined as follows.

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}.$$

Support Vector Machine

Support vector machine (SVM) is another model for binary classification problem and is available in various kernels functions. The objective of an SVM model is to estimate a hyperplane (or decision boundary) based on feature set to classify data points. The dimension of hyperplane varies according to the number of features. As there could be multiple possibilities for a hyperplane to exist in an N-dimensional space

A mathematical representation of the cost function for the SVM model is defined as.

$$J(\theta) = \frac{1}{2}\sum_{j=1}^{n}\theta_j^2 ,$$

Such that

$$\theta^T x^{(i)} \geq 1, \quad y^{(i)} = 1,$$

$$\theta^T x^{(i)} \leq -1, \quad y^{(i)} = 0.$$

3.1. ENSEMBLED LEARNERS

We proposed using existing ensemble techniques along with textual characteristics as feature input to improve the overall accuracy for the purpose of classification between a truthful and a false article. Ensemble learners tend to have higher accuracies, as more than one model is trained using a particular technique to reduce the overall error rate and improve the performance of the model.

Random Forest (RF)

Random forest (RF) is an advanced form of decision trees (DT) which is also a supervised learning model. RF consists of large number of decision trees working individually to predict an outcome of a class where the final prediction is based on a class that received majority votes.

Bagging Ensemble Classifiers

Is an early ensemble method mainly used to reduce the variance (overfitting) over a training set. the bagging model selects the class based on major votes estimated by M number of trees to reduce the overall variance, while the data for each tree is selected using random sampling with replacement from overall dataset. For regression problems, however, the bagging model averages over multiple estimates.

Boosting Ensemble Classifiers

This method allows weak learners to correctly classify data points in an incremental approach that are usually misclassified.

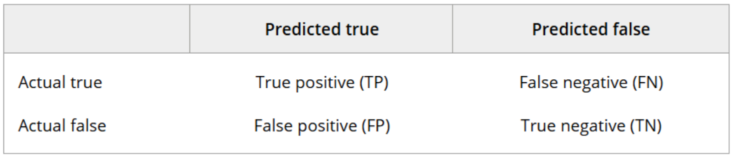
Boosting is another widely used ensemble method to train weak models to become strong learners. For that purpose, a forest of randomized trees is trained, and the final prediction is based on the majority vote outcome from each tree.

Voting Ensemble Classifiers

used for classification problems as it allows the combination of two or more learning models trained on the whole dataset. Each model predicts an outcome for a sample data point which is considered a "vote" in favor of the class that the model has predicted. Once each model predicts the outcome, the final prediction is based on the majority vote for a specific class

4. PERFORMANCE METRICS

To evaluate the performance of algorithms, we used different metrics. Most of them are based on the confusion matrix. Confusion matrix is a tabular representation of a classification model performance on the test set, which consists of four parameters: true positive, false positive, true negative, and false negative.



	Predicted true	Predicted false
Actual true	True positive (TP)	False negative (FN)
Actual false	False positive (FP)	True negative (TN)

Figure 3: Confusion matrix.

Accuracy

Accuracy is often the most used metric representing the percentage of correctly predicted observations, either true or false. To calculate the accuracy of a model performance, the following equation can be used:

$$ACCURACY = \frac{TP + FP}{TP + TN + FP + FN}$$

In most cases, high accuracy value represents a good model.

Precision

precision score represents the ratio of true positives to all events predicted as true. In our case, precision shows the number of articles that are marked as true out of all the positively predicted (true) articles:

$$PRECISION = \frac{TP}{TP + FP}$$

Recall

Recall represents the total number of positive classifications out of true class. In our case, it represents the number of articles predicted as true out of the total number of true articles.

$$RECALL = \frac{TP}{TP + FN}$$

5. RESULTS

Table 1 summarizes the accuracy, precision and recall achieved by each algorithm considered in our dataset.

	Accuracy	Precision	Recall
Logistic Regression (LR)	0.9680	0.9482	0.9821
SVM	0.9840	0.9736	0.9910
Ensemble learners			
Random Forest (RF)	0.9651	0.9347	0.9910
Voting classifier (LR, SVM, RF)	0.9780	0.9590	0.9933
Bagging Classifier (Decision trees)	0.9521	0.9504	0.9419
Boosting Classifier	0.9710	0.9584	0.9776

Table 1: results.

It is evident that the maximum accuracy achieved is 98.40%. achieved by SVM, so in the ensemble learners the maximum accuracy achieved is 97.80%. achieved by Voting classifier.

For the precision is evident that the maximum achieved is 97.36%. achieved by SVM, so in the ensemble learners the maximum accuracy achieved is 95.90%. achieved by Voting classifier.

The last one recall is evident that the maximum achieved is 99.10%. achieved by SVM, so in the ensemble learners the maximum accuracy achieved is 99.33%. achieved by Voting classifier.

6. Conclusion

The task of classifying news manually requires in-depth knowledge of the domain and expertise to identify anomalies in the text.

The learning models were trained and parameter-tuned to obtain optimal accuracy. Some models have achieved comparatively higher accuracy than others.

The primary objective of the research is to identify patterns in text that differentiate fake articles from true news.

The learning models were trained and parameter-tuned to obtain optimal accuracy that is the reason some algorithms were better that others