ANKARA SCIENCE UNIVERSITY, RESEARCHER
VOL. 1, NO. 2, DECEMBER 2021

Detecting Fake News on Big Data

Begüm SUBAŞI[1], **Hilal Nur BERAL**[1], **Nilüfer Nur GÜLEÇ**[1], **Tansel DÖKEROĞLU**[2*]
[1] Department of Computer Engineering, TED University, Ankara, Turkey,
[2] Department of Computer Engineering, Ankara Science University, Turkey, https://orcid.org/0000-0003-1665-5928
*Corresponding Author: tansel.dokeroglu@ankarabilim.edu.tr

Received: 17 August 2021; Revised: 25 August 2021; Accepted: 30 August 2021

Abstract

In this study, we developed a new framework for detecting fake news, which has recently become a significant problem in social media. We compared the performances of different machine learning approaches. It becomes a challenging problem to detect fake news effectively. Apache Spark's machine learning environment, where many processors can work simultaneously, offers a very suitable environment for dealing with big data classification problems. After experiments using Naïve Bayes, Neural Network, Logistic regression, and Support Vector Machine on large datasets we obtained on Kaggle showed that our software can report up to 99% accuracy rates.

Keywords: fake news, machine learning, big data, classification

1. Introduction

Fake news mimics the news media, and it is not based on real information [1]. The false information can spread in a short time to deceive people through social media. The use of fake news increased significantly in 2017 compared to 2016. This type of news is commonly generated in politics. Especially during election times, people tend to share fake news on Twitter. The reason is that they can affect the election result significantly. Although this situation is in favor of some political leaders, it is against others. Fake news is often spread from unverified sources and continues to spread by users in a short time. For these reasons, fake news detection has become a very important issue recently. Experts are slow to inform users of fake news compared to automated fake news detection systems. It is crucial that this fake news, shared in two hours, is detected by automatic detection systems instead of experts.

In this study, we propose a new framework for detecting fake news on social media. We use supervised machine learning algorithms Logistic Regression (LR), Support Vector Machines (SVM), Naive Bayes (NB), Decision Tree (DT), Gradient Boosting (GB), and Random Forest (RF) and compare their performances on benchmark datasets from Kaggle. Since our datasets are big, we use Apache Spark to improve the performance of the algorithms.

Many researchers have studied the methods of fact-checking knowledge. There is an exponential increase of facts created and uploaded on the web every day. Researchers and some journalists commenced creating facts with fake/true news. The purpose is to check the accuracy of those tools. PolitiFact is an internet site that gives quick terms and sentences, fact-checked via a means of journalists. Another famous dataset is FEVER (that includes 185,000 short claims), which constituted Wikipedia sentences. Likely FEVER, the LIAR dataset consists of 13,000 small statements drawn from PolitiFact and classified into six extraordinary classes (pants-fire, fake, barely-true, half-true, mostly-true, true) [2].

Thota et al. used deep learning techniques to locate fake news information in their studies [3]. They achieved 94.21% accuracy in the test data via means of the usage of a finely tuned Dense neural network (DNN) model. According to that, they have obtained 2.5% better results than the current model architectures. They attempted with extraordinary phrase vector illustration and neural network architectures. They accomplished best performing models. Their fine-acting fashions take the TF-IDF vector illustration of phrases with preprocessed engineering capabilities as mixed inputs and use dense neural network structure for predicting the target stance [4].

Nada et al. construct all the classifiers to expect/predict fake news detection. They used an LR classifier. When the model is fit, they compare the f1 score and manage the confusion matrix. The candidate model was decided primarily on the two best models for the fake news classification after all the classifiers were fit. Finally, with the possibility of truth, the chosen model was used for fake news detection. In addition, they extracted the top 50 features from the term-frequency TF-IDF Vectorizer for seeing maximum vital phrases in all the classes. They extensively utilize Precision-Recall and get to know curves for how to train and test datasets are performed after they bloom the amount of data of their classifiers [5].

2. Machine Learning Algorithms

This section gives information about the machine learning techniques used to classify fake news in our study.

2.1 Decision Tree (DT)

DT is a Supervised Machine Learning algorithm in which the data is continuously split according to a specific parameter. It is used where the outcome is a discrete variable like ‘true’ or ‘fake’. A decision tree consists of nodes, branches, and leaf nodes. DT is built on an iterative process of splitting the data. Decision Trees follow Divide-and-Conquer Algorithm. Decision Trees are fast at classifying records, exclude unrelated features, and have a high degree of accuracy comparable to other classification algorithms on many data sets [6].

2.2 Gradient Boosting (GB)

Gradient boosting is an ensemble technique in which the weak learners are converted into strong learners. It combines various weak predictors such as Decision Trees. One of the advantages of the model is that it can handle missing data. In Gradient Boosting, every subsequent predictor learns from previous errors, so the predictions are sequential [7]. Some of the parameters used Gradient Boosting to improve performance such as the number of trees, learning rate, and maximum depth.

2.3 Logistic Regression (LR)

LR provides the intuitive equation to classify problems into binary or multiple classes. It is a model for binary classification problems, and it is used where the output is binary like ‘true’ or ‘fake’, and it works well with the wide feature set. LR makes use of a sigmoid function and transforms the output to a probability value. In this way, it minimizes the cost function to obtain the best probability [8]. It aims to find the most suitable model to describe the relationship between the bidirectional characteristic and a set of related independent variables [9].

2.4 Support Vector Machines (SVM)

SVM is used for binary classification. It is used where the output is like ‘true’ or ‘false’. The SVM model estimates a hyperplane regarding a set of features to classify the data [10]. The dimension of a hyperplane change concerning the number of features. There are many possibilities for a hyperplane to realize in a multi-dimension. The SVM identifies the plane that divides the data with a maximum margin. In our study, we use a linear kernel. Kernels are good to fit the data instances that are not easily separable and/or multidimensional.

2.5 Naive Bayes Algorithm (NB)

It is a supervised classification algorithm based on the Bayes theorem with an assumption of independence among features [11]. The algorithm is “Naive” and works on an assumption that the presence of a feature in a class is independent or unrelated to the other features. The term Naive Bayes is used for classification algorithms based on the theorem of Bayes. Classification algorithms are used to categorize a new observation into previous classes. Bayes Theorem is popular and used to determine the probability of an event based on knowledge-related events [12].

2.6 Random Forest (RF)

RF uses individual decision trees to predict an outcome of a class. The error rate is low compared to other learning methods because of the low correlation of trees [13]. Random Forest is an ensemble algorithm that combines more than one algorithm to classify data. The total number of trees and the decision tree-related parameters like minimum split, split criteria, are the basic parameters of this algorithm [14]. The Gini index is used in our experiments [15] [16].

3. Experimental Setup and Evaluation of the Results

Our fake news detector uses data obtained from Kaggle Twitter Datasets and classifies them as fake or real news. Twitter is one of the most common and important news sources today. We use the SPARK framework and python to classify these big datasets. We test the datasets using machine learning techniques; Naïve Bayer, LR, SVM, Gradient Boosting Classifier, and Decision Tree Classifier. The performance of an algorithm varies with the size and the quality of the text data (or corpus) and the features of the text vectors. Common noisy words called 'stop words' are less important for text feature extraction. They don't contribute to a sentence's actual meaning, but they only contribute to feature dimensionality and may be discarded for better performance. The experiments are carried out on an Intel Core I7-8750 2.20GHz computer with 16 GB RAM. We use 5-fold cross-validation in our experiments. We convert all sentences to lowercase, remove punctuations, stop words, hyperlinks, etc. We don't manipulate the test data. We used the Bag of words method, an NLP technique for feature extraction with the train data. A bag of words is a representation of text that describes the occurrence of words within a document. We keep track of word counts. We selected 1500 of the most frequent features. We use Confusion Matrix and Accuracy values (the number of correct predictions/total number of predictions) to evaluate the performance of the algorithms.

3.1 The Results with Fake News Dataset

In this dataset, there are 20583 tweets and we used 1500 of 155725 features in total. Tables 1 and 2 give the confusion matrix and accuracy percentages of the algorithms, respectively. SVM is the best performing algorithm for this dataset.

Table 1: The Results with Fake News Dataset

	LR	**SVM**	**Gradient Boost**	**Decision Tree**	**Random Forest**	**Naive Bayes**
True positive	1864	2020	2058	2056	1920	1832
False negative	96	102	172	176	384	151
False positive	100	41	25	22	208	229
True negative	1954	2023	1907	1911	1643	1922

Table 2. Accuracy Performance of the Algorithms for Fake News

	LR	**SVM**	**Gradient Boost**	**Decision Tree**	**Random Forest**	**Naive Bayes**
Accuracy %	95	**96**	95	95	85	90

3.2 The Results with True.csv+Fake.csv News Dataset

True.csv and fake.csv datasets contain 44,898 tweets. Tables 3 and 4 give the confusion matrix and accuracy percentages of the algorithms, respectively. LR and SVM are the best performing algorithms for this dataset.

Table 3. The Results with True.csv+Fake.csv News Dataset

	LR	**SVM**	**Random Forest**	**Naive Bayes**
True positive	4611	4620	4725	4678
False negative	14	6	240	203
False positive	17	17	77	124
True negative	4278	4210	4098	4135

Table 4. Accuracy Performance of the Algorithms for with True.csv+Fake.csv News

	LR	**SVM**	**Gradient Boost**	**Decision Tree**
Accuracy %	99	99	96	96

3.3 The Results with Merged News Dataset

In the last step, we merged the Fake News dataset with True.csv and Fake.csv (contains 44.898 tweets) to construct 65,481 tweets in total. Tables 5 and 6 give the confusion matrix and accuracy percentages of the algorithms, respectively. LR is the best performing algorithm for this dataset.

Table 5. The Results of Merged News Dataset

	LR	**SVM**	**Naive Bayes**
True positive	6458	5896	6055
False negative	303	224	953
False positive	289	810	701
True negative	6088	5955	5463

Table 6. Accuracy Performance of the Algorithms for Merged News Dataset

	LR	**SVM**	**Naive Bayes**
Accuracy %	95	92	87

4. Conclusion

This study tested machine learning algorithms, Naïve Bayes, Neural Network, LR, and SVM on big data to classify fake news. We got the highest accuracy (96%) with the SVM model; we got the same accuracy (95%) with Decision Tree, Gradient Boosting, and LR models. In the True.csv+Fake.csv dataset with PYSPARK, we got the highest accuracy (99%) with the SVM and the LR models. We got the same accuracy (96%) with Random Forest and Naive Bayes models. With the biggest dataset merged with the Fake News dataset and True.csv+Fake.csv dataset, we got the highest accuracy (99%) with the LR. So, we can observe that SVM and LR give the best results on the datasets. In future work, we intend to study a deep learning algorithm to classify fake news. A higher-performance computation platform can be used to improve the performance of our framework.

References

[1] Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H., Fake news detection on social media: A data mining perspective. ACM SIGKDD explorations newsletter, 19(1), 22-36, 2017

[2] Nikiforos, M. N., Vergis, S., Stylidou, A., Augoustis, N., Kermanidis, K. L., & Maragoudakis, M., Fake news detection regarding the Hong Kong events from Tweets. In IFIP international Conference on Artificial Intelligence Applications and Innovations (pp. 177-186). Springer, Cham., June 2020.

[3] Thota, A., Tilak, P., Ahluwalia, S., & Lohia, N., Fake news detection: a deep learning approach. SMU Data Science Review, 1(3), 10, 2018

[4] Sahoo, S. R., & Gupta, B. B., Multiple features based approach for automatic fake news detection on social networks using deep learning, Applied Soft Computing, 100, 106983, 2021

[5] Nada, F , Khan, B , Maryam, A , Zuha, N,Ahmed,Z., Fake news detection using logistic regression. International Research Journal of Engineering and Technology (IRJET). https://www.irjet.net/archives/V6/i5/IRJET-V6I5733.pdf , 2019

[6] Medium. “Decision Tree Classification”. https://medium.com/swlh/decision-tree-classification-de64fc4d5aac , Access: 4 June 2021

[7] Towards Data Science. “Gradient Boosting Classification explained through Python”, https://towardsdatascience.com/gradient-boosting-classification-explained-through-python-60cc980eeb3d , Access: 4 June 2021

[8] Ahmad, I., Yousaf, M., Yousaf, S., & Ahmad, M. O., Fake News Detection Using Machine Learning Ensemble Methods. Complexity, 2020.

[9] Erdi, B , Şahin, E , Toydemir, M , Dökeroğlu, T., Makine Öğrenmesi Algoritmaları ile Trol Hesapların Tespiti . Düzce Üniversitesi Bilim ve Teknoloji Dergisi , 9 (1) , 430-442 . DOI: 10.29130/dubited.748366 , 2021

[10] T. Hofmann, B. Schölkopf, and A. J. Smola, “Kernel methods in machine learning,” The Annals of Statistics, vol. 36, no. 3, pp. 1171–1220, 2008.

[11] Medium. “Introduction to Naive Bayes for Classification”, https://medium.com/@srishtisawla/introduction-to-naive-bayes-for-classification-baefefb43a2d , Access: 4 June 2021

[12] Towards Data Science. “Introduction to Naive Bayes Classifier”, https://towardsdatascience.com/introduction-to-naive-bayes-classifier-f5c202c97f92 , Access: 4 June 2021.

[13] B. Gregorutti, B. Michel, and P. Saint-Pierre, “Correlation and variable importance in random forests,” Statistics and Computing, vol. 27, no. 3, pp. 659–678, 2017.

[14] Medium. “Chapter 5: Random Forest Classifier”, https://medium.com/machine-learning-101/chapter-5-random-forest-classifier-56dc7425c3e1, Access: 4 June 2021

[15] L. Breiman, J. Friedman, R. Olshen, and C. Stone, Classification and Regression Trees, Springer, Berlin, Germany, 1984.

[16] Sevinc, E., A novel evolutionary algorithm for data classification problem with extreme learning machines. IEEE Access, 7, 122419-122427, 2019.