*Advances in Parallel Computing Algorithms, Tools and Paradigms*
*D.J. Hemanth et al. (Eds.)*

*doi:10.3233/APC220080*

# Higher Classification of Fake Political News Using Decision Tree Algorithm over Random Forest Algorithm

Dinesh T[a] and RajendranT[b,1]
[a]*Research Scholar, Dept. of CSE, Saveetha School of Engineering,*
[b]*Asso. Prof., Dept. of CSE, Saveetha School of Engineering,*
[a,b]*SIMATS, Chennai, Tamilnadu, India*

**Abstract.** The current project aims to model and compare the performance of fake news detectors using machine learning algorithms to recognize fake news connected to political topics with high accuracy. The Decision Tree algorithm and the Random Forest algorithm are two algorithms. The methods were developed and evaluated on a dataset including 44,000 samples. Implemented each algorithm through programs and performed ten iterations with different scales of false feeds and factual feeds classification were identified. The G-power test is around 80% accurate. For detecting false political news, the Decision Tree algorithm had a mean accuracy of 99.6990, and the Random Forest approach had a mean accuracy of 98.6380, according to the trial results. The significance of accuracy is p=0.001, indicating the efficacy of the classifier. This research aims to use a novel strategy for contemporary Machine Learning Classifiers to predict fake political news. The comparison results reveal that the Decision Tree method outperforms the Random Forest technique.

**Keywords.** Innovative Fake Political News Detection, Decision Tree Algorithm, Random Forest Algorithm, Social Media, Statistics Analytic

## 1. Introduction

This work aims to create a false news tracker that can identify fake political news that is spread on social media [1, 2]. Fake news is not a recent occurrence in the world. For millennia, it has been growing. There were only a few ways to distribute fake news in the past, such as through rumors or social media. However, as the years have passed, social media has become a key venue for news dissemination. Fake news is spreading on social media. Social media is one of the most prevalent mediums for propagating fake news [3]. The findings will enhance false political news prediction, data quality, and implementation [4].

Over 160 Google Scholar papers and 80 IEEE Xplore articles have been published on detecting fake news. This study [5] on social media uses a data mining technique to gather fake news and transform it into a dataset that will be analyzed. It has been referenced 1057 times as a research reference. They analyzed the efficacy of current false news detectors and their limitations in this study [1]. They discussed the existing

[1]Rajendran T, Department of Computer Science and Engineering Saveetha School of Engineering Saveetha Institute of Medical and Technical Sciences, Chennai, Tamil Nadu, India, Email:rajendrant.sse@saveetha.com.

fake news detector's performance and its limitations. To boost the performance, they used neural networks to create a false news detector that was 94.6 percent accurate.

In this study, using a machine learning classifier to identify fake political feeds on social media was more accurate than prior classifiers. The study [6] presented a false news detector that can be used to identify fake news on social media using various sources and classifications. Consider the novel challenge of unsupervised false news identification in this research. Check- It is a web browser extension that swiftly detects fake news while respecting user privacy by assisting with the planning, implementation, and assessment of the project of Check-It [4]. In this study [5], as an example, on social media, a data mining method is used to collect fake news and turn it into a dataset that can be analyzed, which is ideal for future academics interested in detecting fake news. Our diverse research portfolio has resulted in publications in various multidisciplinary initiatives [7]. Now we are focussing on this topic. Our comprehensive portfolio in research has translated into publications in numerous interdisciplinary projects [8] [9].

Previously used methods had a lower accuracy rate, were less trustworthy, and were useless in predicting fake political news. The above observation indicates our ability to recognize fake political news as a result of our study. The study's primary purpose is to enhance false political news classification by introducing new fake news detectors and comparing their performance with machine learning classifiers like the Decision Tree and Random Forest algorithms.

## 2. Materials And Methods

The study was conducted in the Saveetha School of Engineering's CISCO Lab at the SIMATS. Decision Tree and Random Forest are two types of machine learning techniques used; both are supervised models. Execute two rounds on each group using these two algorithms, one for false news detection and the other for accurate news detection. They utilized a programming experiment with N=10 iterations on each approach and ten samples to discover unique scales of fake news and factual news categorization [5]. The G-power test has an average score of roughly 80%. The difference between the two methods is represented by the alpha error rate, which is a type-I error of 0.05. The enrollment ratio is about 1.

### *2.1. Dataset Description*

The "fake_news and actual_news dataset" was employed. The data was gathered using the open forum Kaggle's domain, which is free. This dataset contains data from the 2016 US presidential election. The dataset includes the files "true.csv" and "fake.csv.""Title,""Text,""Subject," and "Date" are the four most significant properties in both files. The text is just required for analysis and classification as a dependent property.

### *2.2. Decision Tree Algorithm*

A supervised learning algorithm is a Decision Tree. It may be used for regression as well as classification. By learning fundamental choice rules from training data, a Choice Tree may be used to develop an innovative model that could be utilized to

forecast the target variable's score. Below are the equations needed to do classification using a Decision Tree (1) Gini index defines the favor more significant probability (2) entropy is used to calculate the homogeneity of the sample (3) information gain is used to compare the samples before and after transformation:

$$Gini = 1 - \sum_{i=1}^{c} 1(p_i)^2 \quad (1)$$

$$\boldsymbol{E(S)} = \sum_{i=1}^{c} 1\text{- } p_i \log_2 p_i \quad (2)$$

$$Information\ Gain = Entropy(before) - \sum_{j=1}^{K} 1Entropy(j, after) \quad (3)$$

**Pseudocode for DTA**

Input: The Collected Dataset

Output: Classifier accuracy is learned

1. The classifier should be fed the training dataset.
2. Create the dtree class.
   dtree is a kind of class.
3. Get all you need from the preceding inputs.
4. Create a new class that will be used to test the attribute.
   def assessment (test attribute)
   in the event that (end loop is leaf)
   return accurate value
   else
   return child[test_attribute]. assessment(test_attribute)
   end
5. Obtain final prediction score(test_attribute).

*2.3. Random Forest Algorithm*

The Random Forest method is a supervised learning approach that may be used to predict and classify data. A Random Forest is a meta classifier that uses averaging to boost predicted accuracy and control over-fitting by fitting several Decision Tree classifiers to different sub-samples of the dataset. If bootstrap='True' is a default value, the subsamples are governed by the max sample size argument; else, every tree is generated using the entire dataset. It is a more advanced variation of the Decision Tree.

**Pseudocode: Random Forest Algorithm**

**Input: The Collected Dataset**

**Output:** Accuracy Prediction value

1. Dataset is loaded as input
2. Randomly, choose 'x' examples from 'b' data.
3. Compute the node 'n' from the 'x' data that use the joint distribution.
4. Nodes are split into child nodes
5. Steps 1 to 3 are repeated until 'l' number of samples is reached
6. The random forest has been built.
7. Compute predict scores with various features.
   prediction_score=rfa_model.predict(set_parameters, "")
8. Score-up for every prediction v is calculated.
9. Obtain a final prediction score

### *2.4. Experiment Setup*

The machine learning methods were tested using the python programming IDE Jupyter lab tool were used. The dataset is made up, but the real news is obtained. The data is preprocessed before being used. Data cleaning is removing non-essential attributes from the dataset, such as title, subject, and date, and concatenating and shuffling them. Through data exploration, the context of the dataset is exposed. Select the data it contains and convert it to the format required by the classifier. The dataset should be divided into two parts: training and testing. The machine learning classifier will be trained with the dataset. The classifier is evaluated using a testing dataset once it has been trained to establish its expected accuracy.

### *2.5. Statistical Analysis*

The SPSS program is used to do statistical calculations such as independent sample T-tests and classifier findings for various test sizes. The Random Forest approach is compared to the Decision Tree algorithm. The critical parameter in the training dataset is 'text', which consists of all kinds of news and is considered the independent variable. The same parameter 'text' has been considered the dependent variable in the testing dataset. The dependent variable is the test data, while the independent variable is the training data.

## 3. Results

The Decision Tree algorithm is around 99 percent accurate, whereas the Random Forest approach is around 98 percent accurate, according to the Accuracy Table (DST, RFA). The accuracy varies depending on the decimal test size. Because of a random variation in the test size, the algorithm's accuracy fluctuates given in Table 1.

**Table 1.** Accuracy Table for DST and RFA obtained with different Test Sizes

| **Test Sizes** | 0.1 | 0.2 | 0.3 | 0.4 |
|---|---|---|---|---|
| **RFA** | 98.88 | 99.06 | 99.08 | 98.75 |
| **DTA** | 99.73 | 99.6 | 99.57 | 99.58 |

The observed statistical values for these two groups based on critical metrics such as mean accurateness and variance for the DTA are 99.699 and 0.10577. The RFA has a score of 98.6380 and a precision of 0.40097. In a statistical examination of ten samples, the Decision Tree had a standard deviation of 0.10577 and a standard error of 0.03345. In contrast, Random Forest had a standard deviation of 0.40097 and a standard error of 0.12680, given in Table 2. Our hypothesis remains true, as evidenced by the significance value of 0.001.

**Table 2.** Group Statistics, the mean precision and standard deviation for DST and RFA.

| | | N | Mean | Std. Deviation | Std. Mean Error |
|---|---|---|---|---|---|
| **Accuracy** | DST | 10 | 99.6990 | 0.10577 | 0.3345 |
| | RFA | 10 | 98.6380 | 0.40097 | 0.12680 |

An Independent Samples Test was performed to evaluate the accurateness of the DTA and the RFA in recognizing false political news, with a significance of 0.001 and a standard error difference of 0.13113. The proposed Decision Tree classifier outperformed the Random Forest classifier when compared to the performance of current methods given in Table 3.

**Table 3**. Independent Sample Test, the correlation of precision for DST and RFA.

| | | Test for Equality of Variances (1) | Test for Equality of Variances (2) | T-test for Equality of Means (3) | T-test for Equality of Means (4) | T-test for Equality of Means (5) |
|---|---|---|---|---|---|---|
| | | **F** | **Sig.** | **Std.Error Difference** | **95% Confidence lower** | **95% Confidence upper** |
| **Accuracy** | **Equal Variances assumed** | 16.440 | 0.001 | 0.13113 | 0.78550 | 1.33650 |
| | **Equal Variances not assumed** | | | 0.13113 | 0.76977 | 1.35223 |

The fake political news detector is the name given to the architecture presented in Figure 1. The design describes the steps needed in developing a false political news detector. Some processes include pre-processing the dataset, data exploration, model classifier, deployment, and accuracy prediction.

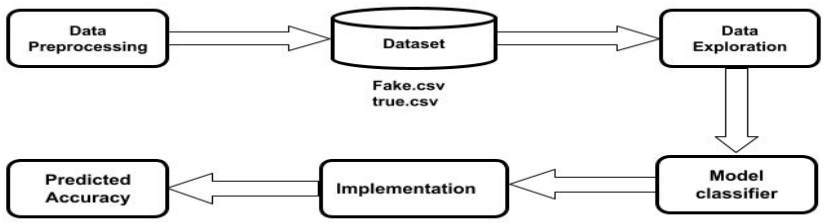


**Figure 1.** Machine learning classifier architecture

In the bar chart, the mean accuracy of the Decision Tree algorithm and the Random Forest approach is 99.6990 and 98.6380, respectively. The Decision Tree method has a 0.03345 error rate, while the Random Forest method has a 0.1268 error rate. The performance of the two algorithms was evaluated using independent - samples t, and a

statistical significance is P=0.001 was observed. The DTA was 99.69 percent accurate, as seen in Figure 2. The proposed Decision Tree classifier outperformed the Random Forest classifier when compared to the performance of current methods.

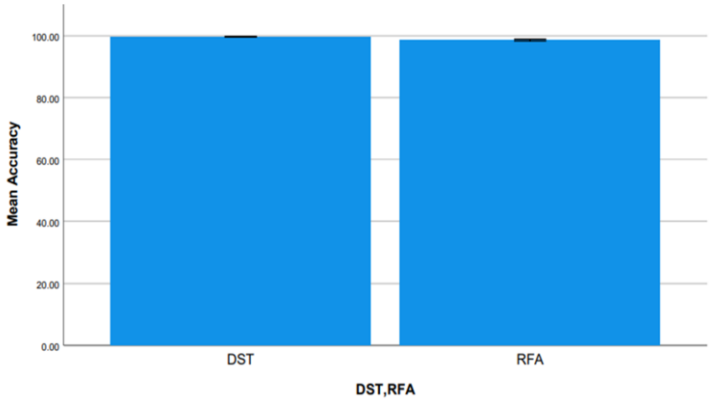


**Figure 2.** Simple Bar Chart depicts the Mean of Accuracy by DST and RFA.

## 4. Discussion

The Decision Tree method outperforms the Random Forest approach in terms of accuracy. Compared to the Random Forest technique, the findings obtained by running numerous rounds of the experiment for identifying the Decision Tree have greater accuracy of roughly 99 percent over 98 percent. When employing the independent samples t-test, the Decision Tree method has a higher significance of 0.001.

The Decision Tree algorithm's mean accuracy and standard deviation are 99.6990 and 0.10577, respectively. From the deep learning methods, some forward-thinking neural networks were utilized to identify the news that falls as fake on social media with 97 percent accuracy [1]. According to the study [10], neural networks perform better in detecting, with an accuracy of 81.92 percent. SVM, on the other hand, has a 91 percent greater accuracy than the Decision Tree, according to [11]. Based on a literature review, it has been established that the Decision Tree method outperforms the naïve Bayes algorithm in terms of accuracy.

Independent sample tests using the SPSS statistical tool demonstrated a statistically significant difference in accuracy between the two algorithms of p0.05. The error rate of DTA is 0.3345, and the Random Forest approach, has an error rate of around 0.12680. With a recall score of 0.942, classifier Decision Trees had the best mistake rate in previous research. With a recall of 0.94, the XGBoost classifier came in second [12].

Our study's main flaw is that only a few indications in the dataset can predict false political news categorization accuracy (%). The higher the number of independent and dependent variables, the better the accuracy. The dataset contains various attributes that the classifier can use to improve prediction accuracy and work more effectively in the future. Using attributes such as profile, source, and evidence, may improve the accuracy and precision of numbers.

## 5. Conclusion

The manual classification of bogus political news necessitates a greater understanding of the area. The difficulty of identifying bogus political news stories using machine learning models was examined in this study. Compared to Random Forest algorithms, the accuracy of a revolutionary false political news detection employing Decision Tree algorithms is higher. The prior study demonstrates that the Decision Tree is more accurate than the classifiers they utilized.

## References

[1] Hiramath C K, Deshpande G C. Fake News Detection Using Deep Learning Techniques. 2019 1st International Conference on Advances in Information Technology (ICAIT). Epub ahead of print 2019. DOI: 10.1109/icait47043.2019.8987258.
[2] Shu K, Mahudeswaran D, Liu H. Fake News Tracker: a tool for fake news collection, detection, and visualization. Computational and Mathematical Organization Theory 2019; 25: 60–71.
[3] Mahyoob M, Algaraady J, Alrahaili M. Linguistic-Based Detection of Fake News in Social Media. DOI: 10.31235/osf.io/umr3t.
[4] Paschalides D, Kornilakis A, Christodoulou C, et al. Check-It: A plugin for Detecting and Reducing the Spread of Fake News and Misinformation on the Web. IEEE/WIC/ACM International Conference on Web Intelligence. Epub ahead of print 2019. DOI: 10.1145/3350546.3352534.
[5] Shu K, Sliva A, Wang S, et al. Fake News Detection on Social Media. ACM SIGKDD Explorations Newsletter 2017; 19: 22–36.
[6] Karimi H, Tang J. Learning Hierarchical Discourse-level Structure for Fake News Detection. Proceedings of the 2019 Conference of the North. Epub ahead of print 2019. DOI: 10.18653/v1/n19-1347.
[7] Sekar D, Lakshmanan G, Mani P, et al. Methylation-dependent circulating microRNA 510 in preeclampsia patients. Hypertens Res 2019; 42: 1647–1648.
[8] Johnson J, Lakshmanan G, M B, et al. Computational identification of MiRNA-7110 from pulmonary arterial hypertension (PAH) ESTs: a new microRNA that links diabetes and PAH. Hypertension Res 2020; 43: 360–362.
[9] Keerthana B, Thenmozhi M S. Occurrence of foramen of huschke and its clinical significance. J Adv Pharm Technol Res 2016; 9: 1835.
[10] Sihombing A, Fong ACM. Fake Review Detection on Yelp Dataset Using Classification Techniques in Machine Learning. 2019 International Conference on contemporary Computing and Informatics (IC3I). Epub ahead of print 2019. DOI: 10.1109/ic3i46837.2019.9055644.
[11] Poddar K, D. GBA, Umadevi KS. Comparison of Various Machine Learning Models for Accurate Detection of Fake News. 2019 Innovations in Power and Advanced Computing Technologies (i-PACT). Epub ahead of print 2019. DOI: 10.1109/i-pact44901.2019.8960044.
[12] Afifah K, Yulita IN, Sarathan I. Sentiment Analysis on Telemedicine App Reviews using XGBoost Classifier. 2021 International Conference on Artificial Intelligence and Big Data Analytics. Epub ahead of print 2021. DOI: 10.1109/icaibda53487.2021.9689735.