International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 06 | June - 2022 Impact Factor: 7.185 ISSN: 2582-3930

NEWS AUTHENTICITY DETECTION USING MACHINE LEARNING ALGORITHMS WITH FEATURE EXTRACTION METHODS

Farjana Farvin S,
Assistant Professor, Department of Computer Science and Engineering, Anjalai Ammal Mahalingam Engineering College, Thiruvarur, India, farzana@aamec.edu.in

Shafinathun Rafiya A,
Student, Department of Computer Science and Engineering, Anjalai Ammal Mahalingam Engineering College Thiruvarur, India, a.rafiya2k@gmail.com

***Abstract*--- In this fast-paced urban technology era, we forgot all about our traditionally used magazines and newspapers and go in search of online news platforms and do not know to what extent those platforms are realistic. Some fake news is spread without knowing the truth or falsehood. In this paper, we put forward a system for detecting news authenticity that uses machine learning algorithms with feature extraction methods. We use Term Frequency- Inverse Document Frequency (TF-IDF) vectorizer, Decision Tree Classifier, Logistic Regression, Gradient Boosting Classifier, Random Forest Classifier. And the experimented outcome shows the great accuracy and precision.**

Keywords*--- *Machine Learning, Decision Tree Classifier, Term Frequency- Inverse Document Frequency (TF-IDF), Logistic Regression, Gradient Boosting Classifier, Random Forest Classifier

INTRODUCTION

Technology development has been very high in the last few years. Needless to say, it is difficult to get through everyday life with us without that technology. We rely on online platforms to find out everything that is going on in the world. So, we believe in everything that comes out of the online platforms not guessing whether it is real or fake and share it with others and that the spread of fake news like this is detrimental to global health. In a post published by the WHO in 2020, it is said that misconceptions about COVID-19 are spreading among the people and therefore people are very scared and confused. Also, it is said that the spread of these fake news has led people to fail to follow the important guidelines set by the WHO and that global health has been severely affected.

In this paper, we are going to manipulate some novel methods to detect the news authenticity that uses a dataset which is available on the Kaggle.com for build a system to identify unreliable news articles. We utilize news headlines helps to determine if the news is real or fake. The obtained data in the datasets are further processed to a splitting of the news headlines as id, title, author, text and label. And the news headlines are interpreted by class 0 and class 1 that is class 0 defines real news and class 1 defines fake news. The collected data undergoes several feature extraction methods such as Data pre-processing and stemming which converts the raw data into a machine understandable data because machine learning algorithms cannot work with raw data. Then the data are split into train data and test data is used to estimate the overall performance of machine learning algorithms when they are used to train the model. And we perform TF-IDF vectorizer that makes use of the frequency or phrases to decide how relevant those phrases are in the given dataset. It's miles pretty easy however intuitive approach to weighting phrases permitting it to behave as a first-rate leaping off point for a ramification of tasks. We made some use of the machine learning algorithm such as Decision Tree Classifier, Random Forest Classifier, Logistic Regression, Gradient Boosting Classifier for the data that obtained to us. These machine learning algorithms are able to gaining knowledge of from the data we provide. As new facts are furnished, the version's accuracy and efficiency to make predictions improve with subsequent training data.

This paper structured as follows: Segment 2 presents some existing approach for news authenticity detection. Segment 3 describes our methodology and its several modules. Segment 4 talks about the execution of our methodology as well as portion of the got results. Segment 5 concludes the paper and presents a few viewpoints**.**

RELATED WORK

In this section we have many literatures related to news authenticity detection.

In [5], authors proposed a simple approach to calculate the frequency and count of news from a set using various machine learning techniques.

In [6], authors proposed a supervised machine learning system to detect fake news in online. They achieved their results by two classification algorithms: Naive Bayes and support Vector Machine.

Authors of [7] present an approach to classifying news based on the title without analyzing the other aspects. The obtained results will be compared with classification based on the whole news data. They analyze the data set and results of news classification of their proposed model

In [8], authors present a framework in order to analyze and discuss the most widely used machine learning techniques. This paper has significantly driven the effort of both academia and industries for developing fake news detection strategies.

In [9], authors proposed a variety of machine learning approaches and used to detect fake and misleading information. They have used a Naive Bayes Classifier, K-Nearest neighbor Classifier for achieving the accuracy.

In [10], authors proposed revision of existing machine learning algorithms such as SVM, Naive Bayes. They provide a great approach for the analyzing and evaluating the classifiers.

Authors of [11] examining well known machine learning algorithms distinctively to validate the efficiency of the classification performance on detecting fake news. They conducted an experiment on widely used public dataset i.e., LIAR, and the results show the efficiency.

Authors of [12] presents an effective automated tool is the necessity to discover such misleading articles. They have done analysis to select the best algorithm which can classify the news article as a real news or fake news.

The general disadvantage of these methodologies is that the straight our information encoding may not be valid. Besides, the standard phony news classification is restricted to two qualities to be specific real or fake, while truly we cannot say that the news is real or fake at 100 percent, however as indicated by a level of certainty. We consider that this point is vital to order the news in online platforms.

METHODOLOGY

The point of our framework is to characterize news validness on news platforms. We portray the system architecture of news authenticity detection system is quite simple and is finished remembering the machine learning algorithms. The system architecture of the proposed system is in the figure1.

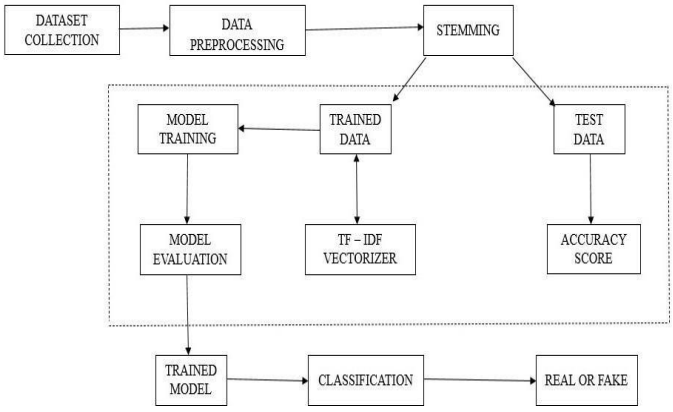

Figure 1. System Architecture

A. DATA PRE-PROCESSING

Data pre-processing is a data mining method which is utilized to change the crude data in a valuable and proficient organization. The nature of the data ought to be checked prior to applying machine learning algorithms. When utilizing data indexes to train machine learning models, you'll frequently hear the expression "garbage in", "garbage out". That's what this intends in the event that you utilize terrible or filthy data to train your model, you'll wind up with a terrible, inappropriately trained model that will not really be pertinent to your examination. Great pre-processed data is much more significant that most remarkable algorithms, to the point that machine learning models prepared with wrong data could really be unsafe to the investigation you're attempting to do – giving you garbage results.

Stop words like as, a, the, an, are, as, at, for are utilized to build sentences. They have no importance whenever utilized as an element in text order. Stop words can be handled and sifted. So, eliminating stop words is the vital stage in NLP. We have utilized Natural Language Toolkit – (NLTK) library to eliminate stop word.

```
import numpy as np
import pandas as pd
import re
from nltk.corpus import stopwords
import nltk
nltk.download('stopwords')
news_dataset = pd.read_csv('/content/train.csv')
news_dataset.shape
news_dataset.isnull().sum()
news_dataset = news_dataset.fillna('')
news_dataset['content'] = news_dataset['author']+' '+news_dataset['title']
X = news_dataset.drop(columns='label', axis=1)
Y = news_dataset['label']
```

B. STEMMING

Stemming is the most common way of diminishing a word to its promise stem that fastens to postfixes and prefixes or to underlying foundations of words known as a lemma. Stemming is significant in Natural Language processing (NLP). Perceiving, looking and recovering more types of words returns more outcomes. At the point when a type of a word is remembered it can make it conceivable to return indexed lists that in any case could have been missed. At the point when another word is found, it can introduce new exploration open doors. Frequently, all that results can be achieved by utilizing the essential morphological type of the word: the lemma. To track down the lemma, stemming is performed by an individual or a calculation, which might be utilized by a machine learning framework. Stemming utilizes various ways to deal with diminish a word to its base from anything that curved structure is experienced. In this, we're using a porter stemmer. It is one of the most famous techniques proposed in 1980. It depends on the possibility that the postfixes in the English language are comprised of a blend of more modest and easier postfixes. This stemmer is known for its speed and clarity.

C. TF- IDF VECTORIZER

TF- IDF represents Term Frequently – Inverse Document Frequency of records. It tends to be characterized as the computation of how important a word in a series or corpus is to a text. The importance expands relatively to the times in the text a word shows up yet is remunerated by the word recurrence in corpus (dataset).

Terminologies:

Term Frequency (TF):

In a document d, the recurrence addresses the quantity of cases of a given word t. The heaviness of a term that happens in a document is essentially corresponding to the term occurrence.

$$tf(t,d) = \frac{Count\ of\ t\ in\ d}{Number\ of\ words\ in\ d}$$

Document Frequency (DF):

This tests the significance of the text, which is basically the same as TF, in the entire corpus assortment n.

$$df(t) = Occurrence\ of\ t\ in\ d$$

Inverse Document Frequency (IDF):

IDF is proportion of how normal or common a term is across the whole corpus of records. So the highlight note is that it's generally expected to every one of the records.

$$idf(t) = log_e[n/df(t)]$$

```
from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer = TfidfVectorizer()
vectorizer.fit(X)
X = vectorizer.transform(X)
```

D. TRAIN DATA & TEST DATA SPLITTING

The train – test split methodology is utilized to evaluate the presentation of machine learning algorithms when they are utilized to make predictions on data not used to train the model. It is a quick and simple methodology to play out, the consequences of which permit you to look at the presentation of machine learning algorithms for your predictive modelling problem.

The data we use is generally parted into training data and test data. The training dataset contains a realized result and model learns on this data to be summed up to different data later on. We have the test dataset (or subset) to test our model's prediction on this subset.

```
from nltk.stem.porter import PorterStemmer
port_stem = PorterStemmer()
def stemming(content):
    stemmed_content = re.sub('[^a-zA-Z]',' ',content)
    stemmed_content = stemmed_content.lower()
    stemmed_content = stemmed_content.split()
    stemmed_content = [port_stem.stem(word) for word in stemmed_content if not word in stopwords.words('english')]
    stemmed_content = ' '.join(stemmed_content)
    return stemmed_content
news_dataset['content'] = news_dataset['content'].apply(stemming)
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.2, stratify=Y, random_state=2)
```

LOGISTIC REGRESSION

Logistic Regression used to foresee the likelihood a prediction has a place with one of two potential classes. We train the model by taking care of it input data and a binary class to which this data has a place. The logistic Regression can be utilized on original input data which the model has never seen (during training). The real or fake classification can be recoded into 0 and 1 for the target variable (machine manage numbers than words). The point of training the logistic Regression model is to sort out the best loads for our direct model inside the Logistic Regression.

```
from sklearn.linear_model import LogisticRegression
model = LogisticRegression()
model.fit(X_train, Y_train)
```

E. DECISION TREE CLASSIFIER

Toward the start, we consider the entire training set as the root. Include values are liked to be categorical. In the event that the qualities are persistent, they are discretized preceding structure the model. Based on credit values are dispersed recursively. We utilize statistical methods for requesting credits as root or the internal node. While going with decision tree, at every node of tree we pose different sort of enquiries. In view of the posed inquiry, we will work out the data gain comparing to it. The consequence of posing an inquiry part the dataset in light of the value of the feature, and makes new nodes. A tree can be learned by parting the source set into subsets in light of attribute value test. This iteration is rehashed on each determined subset in a recursive way called recursive partitioning. The recursion is finished when the subset at a node all has a similar value of the target variable, or while dividing no longer enhances the predictions. The development of decision tree classifier requires no area information or boundary setting and accordingly is proper for exploratory information revelation. Decision trees can deal with high layered data, overall decision tree classifier has great precision. Decision tree induction is a common place inductive way to deal with train data or classification.

```
from sklearn.tree import DecisionTreeClassifier
model1 = DecisionTreeClassifier()
model1.fit(X_train, Y_train)
```

F. GRADIENT BOOSTING CLASSIFIER

Gradient Boosting classifier is a gathering of machine learning algorithms that consolidate numerous frail learning

models together to make areas of strength for a model. It helping models are becoming well known due to their viability at ordering complex datasets. In gradient boosting classifier, every indicator attempts to develop its predecessor by decreasing blunders. However, the captivating thought behind gradient boosting is that is that as opposed to fitting a predictor on the data at every iteration, it really fits another predictor to the remaining mistakes made by the past predictor. For each example in the training set, it computes the residuals for that case or all in all, the noticed value less the anticipated value. Once it has done this, it fabricated another decision tree that really attempts to foresee the residuals that was recently determined. Nonetheless, this is where it gets marginally precarious in examination with gradient boosting classifier.

```
from sklearn.ensemble import GradientBoostingClassifier
model2 = GradientBoostingClassifier()
model2.fit(X_train, Y_train)
```

G. RANDOM FOREST CLASSIFIER

This classifier consolidates various decision trees on various subset of a dataset and midpoints the outcomes to build the dataset's anticipated accuracy. Random forest classifier utilizes ensemble method to come by the ideal outcome. Different decision trees are trained utilizing the training data. This data set contains perceptions and qualities that will be picked in discriminately when nodes are separated. Different decision trees are utilized in a random forest algorithm. It keeps up with great precision even a huge extent of the data is missing. Voting will then, at the point, be performed for each predicted outcome. Lastly, the algorithm will choose the most casted a predicted outcome as the last prediction.

```
from sklearn.ensemble import RandomForestClassifier
model3 = RandomForestClassifier()
model3.fit(X_train, Y_train)
```

IMPLEMENTATION AND RESULTS

The dataset we took is a news dataset. We have disconnected the label column from the data frame. Pre-processing steps are applied and afterward we split the dataset into train and test dataset. Features are extracted using TF- IDF vectorizer.

Table 1. Confusion matrix for the Logistic Regression

Data label	Predicted Real	Predicted Fake
True label	2004	73
Predicted label	14	2069

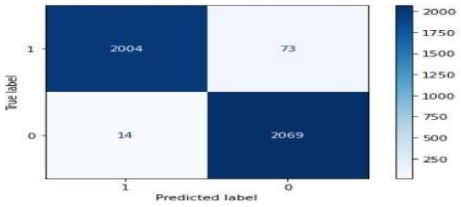

Figure 2 Shows the performance of a classification model on set of trained data and test data.

Figure 2. Performance Matrix

We have created a novel method for Logistic Regression model and gaining the accuracy score for the training data is 0.9865985576923076 and accuracy score for the test data is 0.9790865384615385. The validation of those accuracy score has represented in evaluation graph as shown in the figure 3.

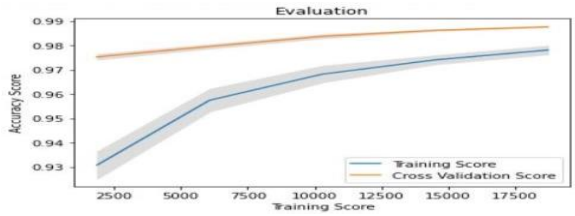

Figure 3. Evaluation Graph

Table 2. Confusion Matrix for Decision Tree Classifier

Data label	Predicted Real	Predicted Fake
True label	2064	13
Predicted label	16	2067

Figure 4 details the performance analysis of the trained data and test data in classification of the trained model, Decision Tree Classifier.

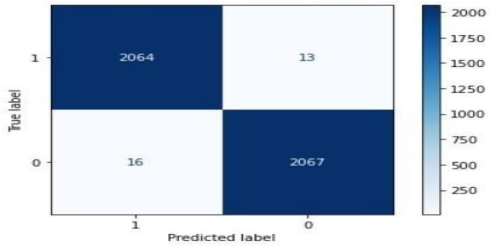

Figure 4. Performance Matrix for Decision Tree Classifier

By the model selection technique of classifying the machine learning algorithm Decision, Tree Classifier has gained the accuracy score for the training data is 1.0 and accuracy score for the test data is 0.9935096615384615. The validation of the obtained accuracy value has been represented in the evaluation graph as shown below in figure 5.

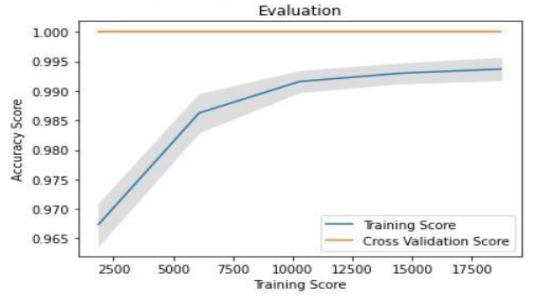

Figure 5. Evaluation Graph for Decision Tree Classifier

Table 3. Confusion Matrix for Gradient Boosting Classifier

Data label	Predicted Real	Predicted Fake
True label	1952	125
Predicted label	10	2073

Figure 6 describes the performance of a classification model, Gradient Boosting Classifier on set of trained data and test data

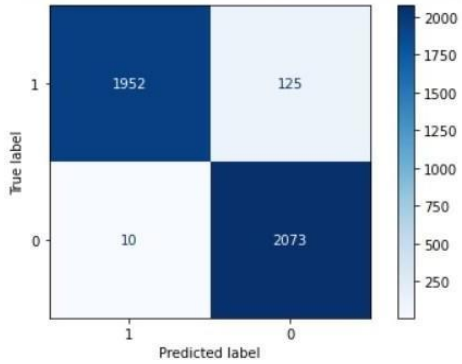

Figure 6. Performance matrix for Gradient Boosting Classifier

Gradient Boosting Classifier defines an efficient way of approaching the model throughout the entire experiment and obtained the accuracy score for the training data is 0.9721153846153846 and the accuracy score for the test data is 0.9675480769230769. Figure 7 shows the graphical representation of obtained validation score of Gradient boosting Classifier.

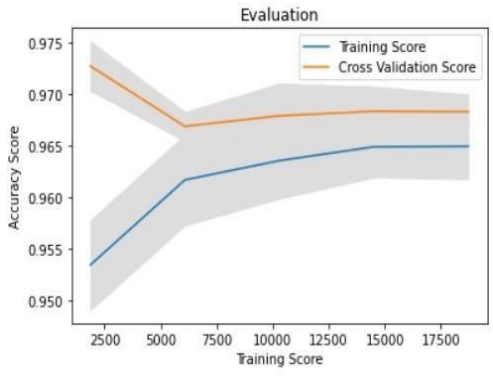

Figure 7. Evaluation Graph for Gradient boosting Classifier.

Table 4. Confusion Matrix for Random Forest Classifier

Data label	Predicted Real	Predicted Fake
True label	2058	19
Predicted label	14	2069

Figure 8 figure out the performance of the trained model, Random Forest Classifier on set of test data and trained data.

By the model selection technique of classifying the machine learning algorithm Random Forest Classifier has gained the accuracy score for the training data is 1.0 and accuracy score for the test data is 0.9939903846153846. The validation of the obtained accuracy value has been represented in the evaluation graph as shown below in figure 9.

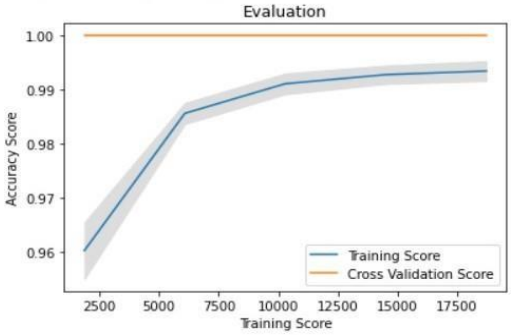

Figure 9. Evaluation Graph of random Forest Classifier

Table 5. Performance Parameter of our Proposed Model

Classification Model	Accuracy Score of Trained Data	Accuracy Score of Trained Data	Overall Percentage of Accuracy Obtained
Logistic Regression	0.9865985576923076	0.9790865384615385	98.2%
Decision Tree Classifier	1.0	0.9935096615384615 4	99.93%
Gradient Boosting Classifier	0.9721153846153846	0.9675480769230769	97.8%
Random Forest Classifier	1.0	0.9939903846153846	99.94%

The above Table 5 details the prediction parameter obtained from our classification models such as Logistic Regression, Decision Tree Classifier, Gradient Boosting Classifier and Random Forest Classifier with different feature extraction methods like data pre-processing, stop words removal, stemming with data and label separation of trained and test data. And most importantly TF- IDF vectorizer play a vital

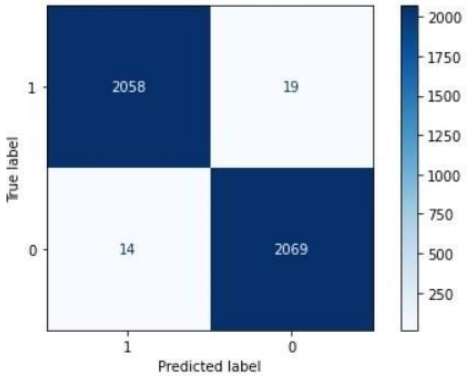

Figure 8. Performance Matrix for Random Forest Classifier part for accomplishing the highest accuracy for the classification models.

e 6. Comparing with other existing models

Classification Models	Accuracy Obtained
Logistic Regression	98.2%
Decision Tree Classifier	99.93%
Gradient Boosting Classifier	97.8%
Random Forest Classifier	99.94%
[6] Marius Cristian Buzea – Naïve Bayes	84.32%
[9] Antonio Galli K - Nearest neighbor Classifier	80.09%
[10] Pranita P. Deshmukh – SVM	87.95%

From the Table 6, we noticed that Random Forest Classifier has the highest Accuracy of 99.94%. We can conclude it by saying our proposed models have the highest accuracy score than the existing models.

CONCLUSION

In this 21st hundred years, most of the assignments are done on the web. Papers that were before liked as printed versions are currently being snubbed by online news platforms. The developing issue of phony news just make things more convoluted and attempts to change or hamper the assessment and disposition of individuals towards computerized innovation. in this way, to control the peculiarity, we have fostered our news authenticity detection framework to group the news headlines as real or fake. In this work, we have presented a disclose model for fake news using feature extraction methods and four diverse machine learning algorithms. However, the earlier concentrate in the news authenticity detection using machine learning algorithms is massive, this work attempts to team up them and contrast the accuracy and various changes of feature extractors. Later on, we expect to foster a more vigorous and more successful model utilizing more rich data so it can extricate includes all more precisely.

REFERENCES

[1] Prof. Dr. Ali Hussein Hasan, Heba Yousef Ateaa. Fake news detection based on the machine learning model. *ResearchGate* [online]. Issue 9,2021:001-9342.

[2] Ms. Kamatam Ashwini, Dr. Birru Devender. Fake news detection using machine learning. *Turkish online journal of Qualitative* [online]. Volume 13, Issue 1, January 2022:408-415.

[3] Development of fake news model using machine learning through NLP. arXiv:2201.07489,2022.

[4] Patel, Pandiya, Singh. Fake news detection using machine learning and natural language processing. 1001:127- 148,2022.

[5] Dr. C K Gomathy, Ms. C V S Vasavi, Mr. D YV Rajesh, Ms. A Srija. The fake news detection using machine learning algorithms. *International Research Journal of Engineering and Technology (IRJET).* Volume 8, Issue 10, October 2021:2395-0056.

[6] Marius Cristian Buzea, Stefan Trausan – Matu, Traian Rabedea. Automatic fake news detection for Romanian online news. *MDPI Information* 2022,13,151. https://doi.org/10.3390/info13030151

[7] Barbara Probierz, Piotr Stefanski, Jankoazk. Rapid detection of fake news based on machine learning methods. *25th International Conference on Knowledge Based and Intelligent Information & Engineering Systems.* Procedia computer science 192(2021)2893-2902.

[8] Antonio Galli, Elio Masciari, Vincenzo Moscato, Giancarto Sperli. A comprehensive benchmark for fake news detection. *Journal of Intelligent Information Systems,* 21 march 2022. https://doi.org/10.1007/s|0844-021-00646-9.Swatipandey, Rashmi Gupta, Jeetendra Kumar. Identification of fake news using machine learning techniques. *Algorithms for Intelligent Systems.* Springer, Singapore 2022. https://doi.org/10.1007/978-981-19-1324-2-25.

[9] Pranita P. Deshmukh, Sakshi A. Dulhani, Parmita C. Aakane, Priyanka Y. Belekar, Isha J Raja. Fake news detection using machine learning. *International journal of research in Engineering Science and Management.* Volume 5, Issue 5, May 2022:2581-5792.

[10] Abdulaziz. Albahr, Marwan Albahr. An empirical Comparison of fake news detection using different machine learning algorithms. *International Journal of Advanced Computer Science and Application (IJACSA).* Volume 11, No.9,2020.

[11] Pooja Malhotra, Sanjay Kumar Malik. Fake news detection using supervised learning techniques. *Journal of Information and Optimization Sciences.* Volume 43, Issue1,2022. https://doi.org/10.1080/02522667.2022.2038933.

[12] Varalakshmi Kongala, Shahana Bano. Fake news

Detection Using deep learning: Supervised fake news detection analysis in social media with semantic similarity method. *IGI global Publisher of Timely Knowledge.* 2022:10.4018/978-1-7998-1192-3.ch011.

[13] Y. B Lasotte, E.J. Garba, Y.M. Malgwi, M.A Buhari. An ensemble Machine learning approach for fake news detection and classification using a soft voting classifier. *European Journal of Electrical Engineering and Computer Science (EJECE).* Volume 6, No.2,2022:2736-5751.

[14] I. Ahamed, M. Yousaf, S. Yousaf, M.O. Ahamed. Fake news detection using machine learning ensemble methods. Complexity (2020). https://doi.org/10.1155/2020/8885861.

[15] Prof. Dr. T. Varpe, Anvita Kulkarni, Rajesh Jadhav, Anoushka Puranik, Meghna Kukreti. Fake news detection using machine learning. *International Journal of advanced Research in Computer and Communication Engineering (IJARCCE).* 2022:2278-1021;10.17148/IJARCCE.2021.10665.

[16] Ghosh, P; Raihan, M; Hassan, M.M; Akter, L; Zaman, S; Awal, M.A. A fake news detection of covid-19 using machine learning techniques. *4th International Conference on Intelligence Computing and Optimization,* IC02021;371:467- 476,2022.

[17] Vookaniti Anurag Reddy, C H Vamsidhar Reddy, Dr. R. Lakshminarayanan. Fake news detection using machine learning. *International Journal for Research in Applied Science & Engineering Technology (IJRASET).* Volume 10, Issue 4, Apr 2022,2321-9653.

[18] Abhi Mitra, Sujan Naha, Debajit Roy, Tanushree Podder. Fake news detection using machine learning. *Abstract of 1st International Conference on Machine Intelligence and System Sciences.* November 2021:978-81-954993-2-8, DOI: 10.21467/abstract.120.

[19] Shalini Pandey, Sankeerthi Prabhakaran, N.V. Subba Reddy, Dinesh Acharya. Fake news detection from online media using machine learning classifiers. *1st International Conference on Artificial Intelligence, Computational Electronics and Communication System.* Volume 2161, issue 1, 2022:10.1088/1742-6596/2161/1/012027.