iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VII **Month of publication:** July 2022

DOI: https://doi.org/10.22214/ijraset.2022.44925

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

Analysis of Classifiers for Fake News Detection: A Machine Learning Perspective

Sasibhushana Rao Pappu[1], N. Sangeetha[2], K. Maniteja[3], Y. Ram Dilip[4], A. Guna Sai Abhishek[5]
[1]*Assistant Professor,* [2,3,4,5] *Under Graduate, Department of Computer Science and Engineering, Aditya Institute of Technology and Management, Srikakulam, Andhra Pradesh, India*

Abstract: ***In the present technology, the usage of the net is increasing day by day. Together with the rise in usage of social media platforms like Facebook, Twitter, etc. The news spread among innumerable people within a brief period. Nowadays all and sundry is using social media accounts for communication with one another like within the regions like Political events, Technological area, movie events, etc.....***
Some people on social media spread the fake news in an immovable way. Nowadays it's become a troublesome job to detect fake news on social media. Because the platform doesn't verify the activities of users and their posts is difficult to acknowledge that fake news is either a positive thing or a negative thing. As a personality being, it's tough to detect all this fake news. Solution for this work.
Our study explores different textual properties which will be wont to distinguish fake content from real. By using those properties, we train the info by using the various Machine learning algorithms like SVM, Multinomial Naive Bayes, and Random Forest by using The TF-IDF vectorizer to judge their performance on 2 datasets. We aim to supply the user with the flexibility to classify the news as fake or real. The experimental evaluation confirms the superior performance of our proposed ensemble learner approach compared to individual learners. While comparing the three algorithms Multinomial Naive Bayes, Linear Support Vector Machine, and Random Forest. We conclude that the Support Vector Machine Algorithm gives more accuracy than the opposite two algorithms.
Index Terms: Social Media, Fake News, Classification of News, TF-IDF Vectorizer, Multinomial Naive Bayes, Linear SVM, Random Forest

I. INTRODUCTION

Fake news refers to information content that's false, misleading, or whose source cannot be verified. This content is also generated to intentionally damage reputations, deceive, or gain attention Clickbait, Satire/parody, Propaganda, Biased, and Unreliable news are various varieties of fake news The technological simple copying, pasting, clicking, and sharing content online has helped these forms of articles to proliferate. In some cases, the articles are designed to electrify an emotional response and placed on certain sites ("seeded") to entice readers into sharing them widely. In other cases, "fake news" articles are also generated and disseminated by "bots" - computer algorithms that are designed to act like people sharing information but can do so quickly and automatically.
Manual fake news detection often involves all the techniques and procedures someone can use to verify the news. It could involve visiting fact-checking sites.
It might be crowdsourcing real news to match with unverified news. But, the number of information generated online daily is overwhelming. Also noting how briskly information spreads online, manual fact-checking quickly becomes ineffective. Manual fact-checking struggles to scale with the degree of knowledge generated, Therefore, highlighting the explanation behind the creation of automated fake news detection. Automated detection systems provide value in terms of automation and scalability. There are various techniques and approaches implemented in fake news detection research. And it's worth noting that these approaches often overlap counting on perspective.

II. RELATED WORK

Akshay Jain and Amey Kasbe in their paper [1] show a simple approach for fake news detection using a naive Bayes classifier. They have tested the difference in accuracy by taking the different lengths of articles for detecting the fake news and they also used the concept of web scrapping was introduced which gave us an insight into how we can update our dataset on regular basis to check the truthfulness of the recently updated Facebook posts. They get the dataset from the Github containing 11000 news articles tagged as real or fake.

Syed Ishfaq Manzoor, Dr. Jimmy Singla, and Nikita in their page [2] this paper reviews various Machine learning approaches in the detection of fake and fabricated news.

The limitation of such approaches and improvisation by way of implementing deep learning is also reviewed. Iftikhar Ahmad, Muhammad Yousaf, Suhail Yousaf, and Muhammad Ovais Ahmad are authors [6] on this page, Fake News Detection Using Machine Learning Ensemble Methods. They extracted different textual features from the articles using an LIWC tool and used the feature set as an input to the models.

Nihel Fatima Baarir and Abdelhamid Djeffal [3], Fake News detection Using Machine Learning using Support Vector Machine, TF-IDF of a bag of words and n-grams as a featureextraction technique. They have merged two existing datasets "Getting Real about FakeNews" which contains fake news and "All the news" containing real news. These datasets were get from the Kaggle site. They get an accuracy of 82%.

Uma Sharma, Sidarth Saran, Shankar M. Patil by this author [4], Fake News Detection using Machine Learning Algorithms using Logistic Regression in two ways are static search andgrid search. It aims to provide the user with the ability to classify the news as fake or real and also check the authenticity of the website publishing the news.

They get the data from online news from different sources like social media websites, search engines, For example static search,our best model came out to be Logistic Regression with an accuracy of 65%. Hence we thenused grid search parameter optimization to increase the performance of logistic regression which then gave us an accuracy of 75%. The accuracy for dynamic the system is 93% and it increases with every iteration.

Anjali Jain, Harsh Khatter, and AvinashShakya, authors [5] in this page, "a smart system for fake news detection using machine learning" using Support Vector Machine (SVM), NaiveBased Classifier, NLP. It is based on the idea of finding the hyper-plane that best divides thedataset into two classes. They get an accuracy of approximately 93% by using the three classifiers.

Iftikhar Ahmad, Muhammad Yousaf, Suhail Yousaf, and Muhammad Ovais Ahmad are authors [6] on this page, Fake News Detection Using Machine Learning Ensemble Methods.

They extracted different textual features from the articles using an LIWC tool and used the featureset as an input to the models.They use the three datasets are the first dataset is called the "ISOT Fake News Dataset", and the second and third dataset is available at Kaggle.

Using these datasets on different classifiers like Logistic regression (LR), Linear SVM (LSVM), Multilayer perceptron, K-nearest neighbors(KNN), Random Forest (RF), and Decision tree. As compared to all methods highest average accuracy was getting in the decision tree is 91% by using all three datasets.

Okuhle Ngada and Bertram Haskins authors [7] in this page, Fake News Detection Using Content-Based Features and Machine Learning. Natural Language Processing, Text Analysis, and Support Vector Machine classifiers are used in this paper. The Kaggle-hosted dataset consists of two datasets; one fake news dataset consisting of 23 481 fake news articles, and a real news dataset containing 21 417 real news articles. The datasets contain articles published between the years 2015 and 2018.

Vivek Singh, Rupanjal Dasgupta, Darshan Sonagra, and Karthik Raman are the authors [8] of this page. Text Processing, Support Vector Machine methods are used. "Kaggle Fake News" dataset as provided by the SBP-BRIMS organizers. They used LIWC (Linguistic Analysis and Word Count) package to obtain linguistic features for each of the articles. Creation of a text-processing-based machine learning for automatic identification of Fake News with 87%accuracy.

Ankit Kesarwani, Sudakar Singh Chauhan, and Anil Ramachandran Nair are the authors [9]of this page. K-Nearest Neighbor; Data Mining; Supervised are used in this paper. The dataset has been collected from Buzz Feed News organization that was used in the training andtesting of the model. Buzz Feed News used social analytic services Buzz Sumo to identify the top-performing Facebook content from 167 websites that consistently publish the articles. The approach achieved a maximum classification accuracy of 79% using KNN.

Mykhailo Granik and Volodymyr Mesyura are the authors [10] of this page. Naive Bayes classifier; artificial intelligence is used in their page. Dataset, collected by Buzz Feed News, was used for learning and testing the naive Bayes classifier. The dataset contains informationabout Facebook posts, each of which represents a news article.

They were collected from three large Facebook pages each from the right and from the left, as well as three large mainstream political news pages (Politico, CNN, and ABC News). They approach isachieved a maximum classification accuracy of a 75.4% using naive Bayes classifier.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

III. METHODOLOGY

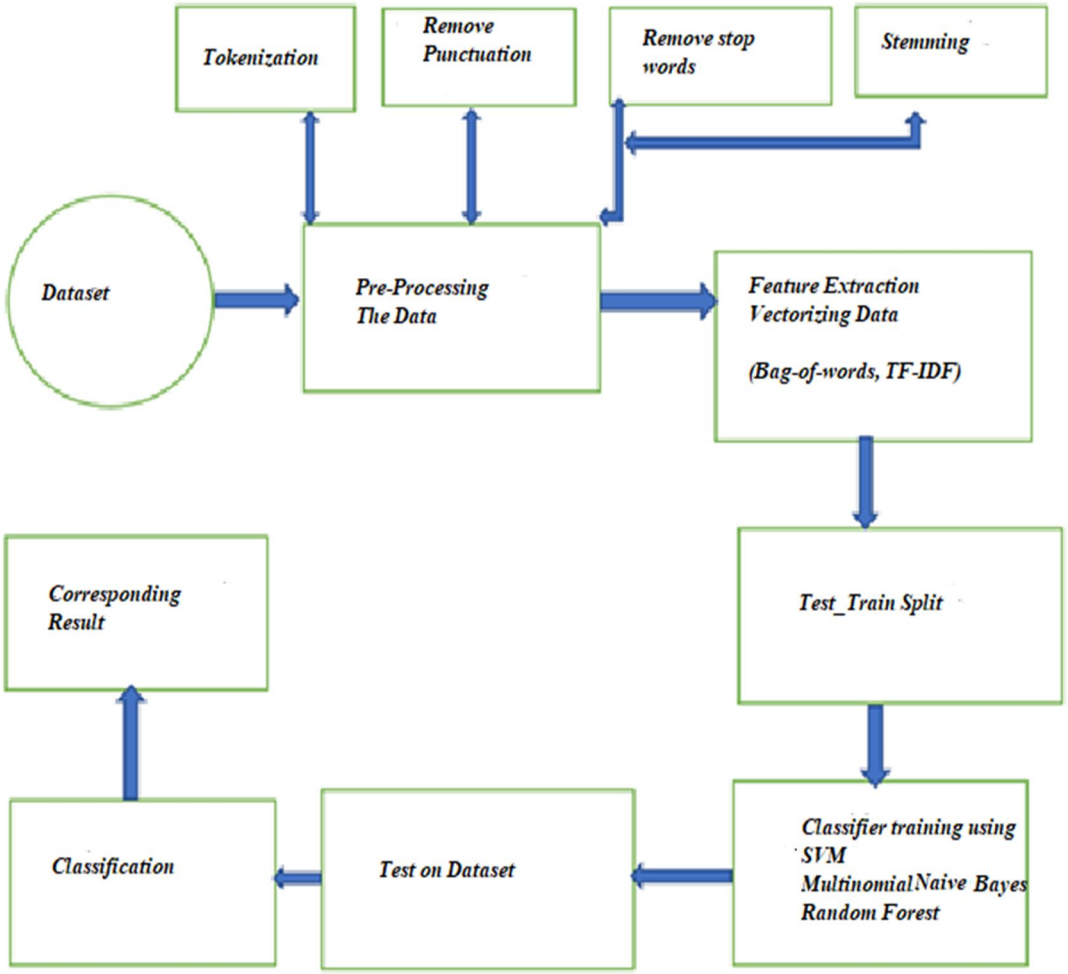


Fig Proposed System Methodology

The dataset was obtained from Kaggle. Here we are taking two datasets i.e. Fake.csv and True.csv. The Fake.csv dataset consists of 23504 rows and 4 columns i.e. Title, text, subject and the date and here subject consists of political News, World News, News, Politics, Government News, Left_news and US_news and the True.csv dataset consists of 21418 rows and 4 columns i.e. Title, text, subject, date. Here subject consists of political News, World News, News, Politics, Government News, Left_news and US_news. In this we are storing the fake news dataset in a fake data frame and the real news dataset in a real data frame after that we are displaying the first five rows in real and fake dataframes after that we are creating a column label in both real and fake data frames, for realwe are inserting ones for label column and for fake data frame we are inserting zeroes andconcatenate both these data frames to df by using pd.concat. Then visualize the class distribution so that the data set is balanced.

We use some preprocessing techniques to calculate the missing values and remove the duplicates from the data and remove the duplicate entries in the text column. We also remove the stopwords and punctuations in the data. We apply feature extraction techniques like TF-IDF vectorization later we train the model with different classification algorithms on the data. The algorithms like Linear Support Vector Machine, Multinomial Naive Bayes and Random Forest. Here we apply three cases on the data set Case-1: Both Title and Text, Case-2: only title and Case-3: only text and we apply three algorithms like Multinomial Naive Bayes, Linear Support Vector Machine and Random Forest.

A. Algorithms Used

Support Vector Machine

A support vector machine is in a way a binary classifier since the model works by generating a hyperplane that's accustomed separate the training data as far as possible. The support vector machine performs well since there's an especially high number of features in a very text classification problem but generally requires plenty of tuning and is memory intensive. Once we've labeled training data (supervised learning), the algorithm generates the simplest possible hyperplane which categorizes new data automatically. In an exceedingly two-dimensional space, this hyperplane would be a line divided into a plane of two Parts with each class lying on either side

B. Types Of SVM

SVM is of two types

1) *Linear SVM:* Linear SVM is employed for linearly separable data, which implies if a dataset may be classified into two classes by employing a single line, then such data is termed linearly separable data, and classifier is employed called Linear SVM classifier.
2) *Non-linear SVM:* Non-Linear SVM is employed for non-linearly separated data, which suggests if a dataset cannot be classified by employing a line, then such data is termed non-linear data, and the classifier used is termed Non-linear SVM classifier.

C. Multinomial Naive Bayes

Multinomial Naive Bayes algorithm could be a probabilistic learning method that's mostly employed in tongue Processing (NLP). The algorithm relies on the Bayes theorem and predicts the tag of a text like a chunk of email or news article. It calculates the probability of every tag for a given sample then gives the tag with the best probability as output. A Naive Bayes classifier may be a collection of the many algorithms where all the algorithms share one common principle, which is each feature being classified isn't associated with the other feature. The presence or absence of a feature doesn't affect the presence or absence of the opposite feature.

D. How Multinomial Naive Bayes works?

Naive Bayes could be a powerful algorithm that's used for text data analysis and with problems with multiple Classes. To know Naive Bayes theorem's working, it's important to know the Bayes theorem concept first because it is predicated on the latter. Bayes theorem, formulated by Bayes, calculates the probability of a happening occurring supported the prior knowledge of conditions associated with an occasion. It's supported the subsequent formula:

` **P (A|B) = P (A) * P (B|A) / P (B)**

Where we are calculating the probability of class A when predictor B is already provided.
P (B) = prior probability of B
P A) = prior probability of class A
P (B|A) = occurrence of predictor B given class A probability
This formula helps in calculating the probability of the tags within the text.

E. Random Forest

As the name suggests, the Random Forest algorithm generates the forest with a variety of decision trees. So it's the gathering of decision trees. Decision trees are attractive classifiers among others due to their high swiftness. Supported random samples from the database a random forest classifier averages multiple decision trees. Generally, the more trees within the forest are that the sign that the forest is powerful. Similarly, within the random forest classifier, high accuracy is obtained by higher the number of trees within the forest. While concurrently creating a tree with decision nodes, a call tree breaks the dataset down into smaller subsets. The choice root node is that the selected through highest information gain and leaf nodes supported a pure subset for every iteration simultaneously. Calculation of knowledge Gain (IG) requires impurity measure (Entropy) of that node. There are various indices to live the degree of impurity. A leaf node represents a category or pure subset. The trees in a random forest are created under random data so there may well change to be a lack of meaning and noise. So as to form a model with low variance random forest averages these trees. The irrelevant trees drop one another out and therefore the staying meaningful trees yield the ultimate result.

IV. RESULTS

A. *Case-1: (While considering both title and text)*

1) *Multinomial Naive Bayes*

```
Confusion Matrix :
 [[3786  587]
 [ 121 5168]]
Classification Report :
              precision    recall  f1-score   support

           0       0.97      0.87      0.91      4373
           1       0.90      0.98      0.94      5289

    accuracy                           0.93      9662
   macro avg       0.93      0.92      0.93      9662
weighted avg       0.93      0.93      0.93      9662
```

Figure 1 Case-1 Multinomial Naive Bayes Classification Report

2) *Linear Support Vector Machine*

```
Confusion Matrix :
 [[4274   45]
 [  24 5319]]
Classification Report :
              precision    recall  f1-score   support

           0       0.99      0.99      0.99      4319
           1       0.99      1.00      0.99      5343

    accuracy                           0.99      9662
   macro avg       0.99      0.99      0.99      9662
weighted avg       0.99      0.99      0.99      9662
```

Figure 2 Case-1 Linear Support Vector Machine Classification Report

3) *Random Forest*

```
Confusion Matrix :
 [[4216  103]
 [  38 5305]]
Classification Report :
              precision    recall  f1-score   support

           0       0.99      0.98      0.98      4319
           1       0.98      0.99      0.99      5343

    accuracy                           0.99      9662
   macro avg       0.99      0.98      0.99      9662
weighted avg       0.99      0.99      0.99      9662
```

Figure 3 Case-1 Random Forest Classification Report

B. CASE-2 (While considering only title)

1) Multinomial Naive Bayes

```
Confusion Matrix :
 [[4026  336]
 [ 271 5029]]
Classification Report :
              precision    recall  f1-score   support

           0       0.94      0.92      0.93      4362
           1       0.94      0.95      0.94      5300

    accuracy                           0.94      9662
   macro avg       0.94      0.94      0.94      9662
weighted avg       0.94      0.94      0.94      9662
```

Figure 4 Case-2 Multinomial Naive Bayes Classification Report

2) Linear Support Vector Machine

```
Confusion Matrix :
 [[4046  316]
 [ 208 5092]]
Classification Report :
              precision    recall  f1-score   support

           0       0.95      0.93      0.94      4362
           1       0.94      0.96      0.95      5300

    accuracy                           0.95      9662
   macro avg       0.95      0.94      0.95      9662
weighted avg       0.95      0.95      0.95      9662
```

Figure 5 Case-2 Linear Support Vector Machine Classification Report

3) Random Forest

```
Confusion Matrix :
 [[3885  477]
 [ 199 5101]]
Classification Report :
              precision    recall  f1-score   support

           0       0.95      0.89      0.92      4362
           1       0.91      0.96      0.94      5300

    accuracy                           0.93      9662
   macro avg       0.93      0.93      0.93      9662
weighted avg       0.93      0.93      0.93      9662
```

Figure 6 Case-2 Random Forest Classification Report

C. CASE-3 (While considering only text)

1) Multinomial Naive Bayes

```
Confusion Matrix :
 [[3691  700]
 [  66 5205]]
Classification Report :
              precision    recall  f1-score   support

           0       0.98      0.84      0.91      4391
           1       0.88      0.99      0.93      5271

    accuracy                           0.92      9662
   macro avg       0.93      0.91      0.92      9662
weighted avg       0.93      0.92      0.92      9662
```

Figure 7 Case-3 Multinomial Naive Bayes Classification Report

2) Linear Support Vector Machine

```
Confusion Matrix :
 [[4343   48]
 [  22 5249]]
Classification Report :
              precision    recall  f1-score   support

           0       0.99      0.99      0.99      4391
           1       0.99      1.00      0.99      5271

    accuracy                           0.99      9662
   macro avg       0.99      0.99      0.99      9662
weighted avg       0.99      0.99      0.99      9662
```

Figure 8 Case-3 Linear Support Vector Machine Classification Report

3) Random Forest

```
Confusion Matrix :
 [[4282  109]
 [  28 5243]]
Classification Report :
              precision    recall  f1-score   support

           0       0.99      0.98      0.98      4391
           1       0.98      0.99      0.99      5271

    accuracy                           0.99      9662
   macro avg       0.99      0.98      0.99      9662
weighted avg       0.99      0.99      0.99      9662
```

Figure 9 Case-3 Random Forest Classification Report

4) Testing Accuracy

	Case-1 (Both Title and text)	Case-2 (only title)	Case-3 (only text)
Multinomial Naïve Bayes	0.9268	0.9371	0.9207
Random Forest	0.9854	0.93	0.9858
Support Vector Machine	0.9928	0.9457	0.9927

Fig 6.1 Testing Accuracy

Here we apply three cases on the data set Case-1: Both Title and text, Case-2: only title,and Case-3: only text and we apply three algorithms like Multinomial Naive Bayes, Support Vector Machine and Random Forest.

V. CONCLUSION

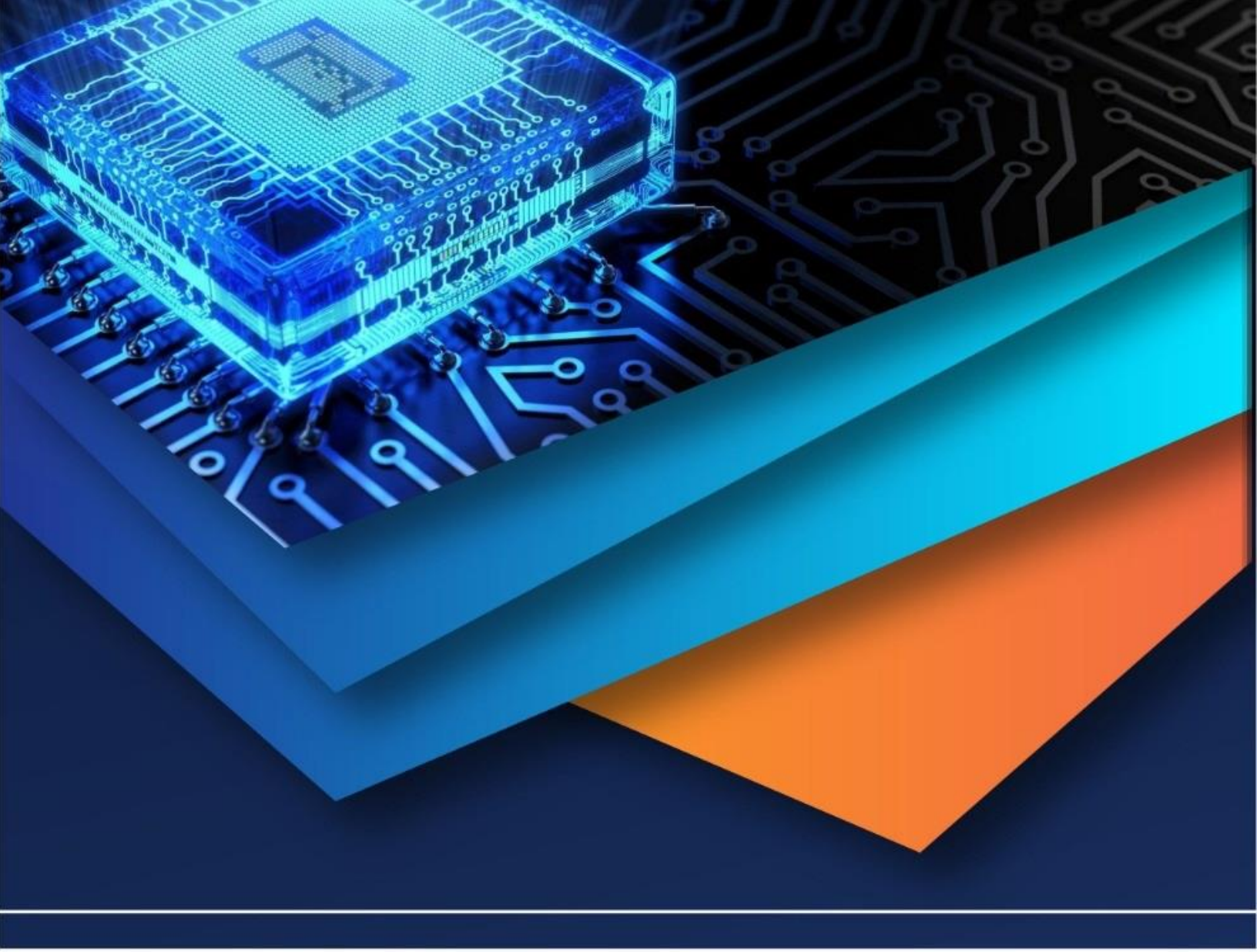
Due to the increasing use of the net, it's now easy to spread fake news. A large number of individuals are regularly connected with the net and social media platforms. There's no restriction while posting any news on these platforms. So a number of the people take the advantage of those platforms and begin spreading fake news against the individual organizations. This could destroy the repute of a private or can affect a business. Through fake news, the opinion of the people can even be changed for a party. There's a necessity for some way to detect this fake news. In this project we use feature extraction methods like TF-IDF Vectorizer and also different classification algorithms like Linear Support Vector Machine classifier, Multinomial Naive Bayes classifier and Random Forest classifier are accustomed classify the news as real or fake. The classifiers are first trained with an information set called training data set. After that, these classifiers can automatically detect fake news. Based on obtaining the test accuracy, while comparing the three algorithms Multinomial Naive Bayes, Linear Support Vector Machine, and Random Forest. We conclude that the Support Vector Machine Algorithm gives more accuracy than the opposite two algorithms.

VI. FUTURE SCOPE

In this project, we implemented machine learning classifiers with both bag-of-words and TF-IDF vectorization for converting text into numeric vectors. Using these, we cannot provide semantic relations between words. Utilizing transformers to provide contextual embedding is the future era on this topic. Deep learning algorithms in different perspectives like hybrid models and ensemble approaches are also to be considered.

REFERENCES

[1] Akshay Jain and Amey Kasbe, "Fake news detection using naive Bayes classifier", 2018 IEEE International Students' Conference on Electrical, Electronics and Computer Sciences, 978-1-5386-2663-4, 2018.
[2] Syed Ishfaq Manzoor, Dr. Jimmy Singla and Nikita, "Fake News Detection Using Machine Learning approaches: A systematic review", Proceedings of the Third International Conference on Trends in Electronics and Informatics (ICOEI 2019) IEEE, ISBN: 978-1-5386-9439-8.
[3] Nihel Fatima Baarir and Abdelhamid Djeffal , "Fake News detection Using Machine Learning", 2020 2nd International Workshop on Human-Centric Smart Environments for Health and Well-being (IHSH), 978-1-6654-4084-4/21/$31.00 ©2021 IEEE.
[4] Uma Sharma, Sidarth Saran, Shankar M. Patil, "Fake News Detection using Machine Learning Algorithms", Department of Information Technology Bharati Vidyapeeth College of Engineering Navi Mumbai, India, International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181, 2021.
[5] Anjali Jain, Harsh Khatter and AvinashShakya, "A SMART SYSTEM FOR FAKE NEWS DETECTION USING MACHINE LEARNING", Department of Computer Science & Engineering, ABES Engineering College Dr. APJ Abdul Kalam University, Lucknow, India, International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)46931.2019.8977659, September 2019.
[6] Iftikhar Ahmad, Muhammad Yousaf, Suhail Yousaf, and Muhammad Ovais Ahmad, "Fake News Detection Using Machine Learning Ensemble Methods", Department of Computer Science and Information Technology, University of Engineering and Technology, Peshawar. Hindawi Complexity Volume 2020, Article ID 8885861, 2020. https://doi.org/10.1155/2020/8885861.
[7] Okuhle Ngada and Bertram Haskins, "Fake News Detection Using Content-Based Features and Machine Learning", School of IT Nelson Mandela University Port Elizabeth, South Africa. 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE) | 978-1-6654-1974-1/20/$31.00 ©2020 IEEE | DOI: 10.1109/CSDE50874.2020.9411638.
[8] Vivek Singh, Rupanjal Dasgupta, Darshan Sonagra, Karthik Raman, and Isha Ghosh, "Automated Fake News Detection Using Linguistic Analysis and Machine Learning", Behavioral Informatics Lab, Rutgers University, New Brunswick, NJ 08901. All content following this page was uploaded by Isha Ghosh on 19 July 2017.
[9] Ankit Kesarwani, Sudakar Singh Chauhan and Anil Ramachandran Nair, "Fake News Detection on Social Media using K-Nearest Neighbor Classifier". Department of ECE National Institute of Technology Kurukshetra, India. 978-1-7281-6362-8/20/$31.00 ©2020 IEEE.
[10] Mykhailo Granik, Volodymyr Mesyura, "Fake News Detection Using Naive Bayes Classifier". Computer Science Department Vinnytsia National Technical University Vinnytsia, Ukraine. 978-1-5090-3006-4/17/$31.00 ©2017 IEE



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY