2023 3rd International Conference on Intelligent Technologies (CONIT)
Karnataka, India. June 23-25, 2023

Comparative Analysis Of Machine Learning Models For Fake News Classification

Archit Gupta
Department of Applied Mathematics
Delhi Technological University
New Delhi, India
architgupta1202@gmail.com

Arnav Batla
Department of Applied Mathematics
Delhi Technological University
New Delhi, India
batlaarnav@gmail.com

Chaitanya Kumar
Department of Applied Mathematics
Delhi Technological University
New Delhi, India
kumarchaitanya1108@gmail.com

Dr. Goonjan Jain
Department of Applied Mathematics
Delhi Technological University
New Delhi, India
goonjanjain@dtu.ac.in

2023 3rd International Conference on Intelligent Technologies (CONIT) | 979-8-3503-3860-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/CONIT59222.2023.10205870

***Abstract*—It has become clear that fake news is dangerous. Identifying fake news is a crucial step towards preserving the virtue and prosperity of society. Social media's rising popularity has led to an increase in the spread of false information. There aren't enough frameworks in place to deal with misleading news. There are many low-cost online news sources and it's an easy access via social media. These are the reasons why there's a spread of fake news. News Content is the only reason for the present fake news detection algorithms, also users' previous posts or activities provide a lot of insights about their views on news and have a significant effect on false news identification. The proposed research seeks to investigate several machine learning approaches for the analysis and identification of false news. In order to identify the spread of fake news on social media, we compare various widely used machine learning methods, such as Naive Bayes and Multi Layer Perceptron Classifiers, in this study. In this work, using solely text data, we develop a number of machine learning methods using the WELFake dataset.**

***Index Terms*—fake news detection, machine learning, text analysis, naive bayes, multi layer perceptron**

I. Introduction

Nowadays, social media networks are the most popular way to distribute thousands of pieces of news, whether they are public or private. Content on social media is simple to obtain, share, and remark on. There is a danger of getting exposed to false information which is purposefully wrong or contains inaccurate information in order to support a certain political or economic objective as it is simpler for users or readers to share their particular opinions. Furthermore, false information typically spreads more quickly, deeply, and widely on social media.

In the past year, fake news detection in social media has drawn a lot of attention from academic and professional circles. Social media networks have identified numerous web pages and misinformation and have devoted resources to the job. A fake news detection method aims to identify and investigate various types of potentially misleading news. The analysis of prior real and fake news samples is used to predict the likelihood that a given piece of news content is being intentionally misleading.

A notable obstacle to this algorithmic method and resolving this natural language processing issue is the lack of news examples that are available as corpora for training the model. The various false news models that already existed were context-specific. There is a dearth of a suitable paradigm for classifying the different kinds of deceptions that can happen when working with written data. This study looks at various techniques and kinds of deception that can be found when working with online news material, weighing the benefits and drawbacks of predictive modeling. It offers an algorithmic method for resolving the specified issue.

We used the WELFake dataset, which consists of 72,134 news stories, 35,028 of which are true and 37,106 of which are false. In order to avoid over-fitting of classifiers and to give more text data for better ML training, authors combined four well-known news datasets (Kaggle, McIntire, Reuters, and BuzzFeed Politics). The four columns in this dataset are Serial number (beginning at 0), Title (about the text news headline), Text (about the news substance), and Label (0 = fake, 1 = authentic). Just 72134 of the 78098 data elements in the csv file are accessible according to the data frame.

The primary goal of this study is to minimise and decrease the spread of false information on social media platforms. It is more practical for the consumer when the virtual site is secure. In this research, we'll attempt to put some methods into practise to see how well the algorithms performs at detecting authenticity. We think it will aid in the identification of rumours and have a transformative effect on the veracity of social media news. Additionally, it will stop people from trusting anything they read or see on social media.

II. Related Works

Akshay Jain and Amey Kasbe conducted a research study in which they evaluated the accuracy of various machine learning algorithms for making predictions. They tested models such as

bounded decision trees, gradient boosting, and support vector machines, using an unreliable probability threshold. The study found that these models achieved an accuracy ranging between 0.85 to 0.91.[1]

In their study, Farzana Islam and her colleagues utilized the Naïve Bayes classifier algorithm method to categorize deceptive news, employing both count vectorization and TF-IDF vectorization as feature extraction methods. They applied this technique to two open source fake news datasets available on various sources[2].

Zongru Shao and Pranav Bharadwaj conducted experiments on the fake news dataset available on Kaggle.com, utilizing RNN, , a random forest classifiers and a Naïve Bayes classifier. They employed various feature extraction techniques, including Quadgram, Bigram, Trigram, and GloVe[3].

Castillo et al. used a different and more logical approach in their research. They identified positive and negative keywords and hashtags to identify the legitimacy of the news[4].Karimi and Tang introduced a novel technique to detect fabricated news, which relies on HDSF more commonly known as Hierarchical Discourse-level Structure. It is a hierarchical structure. In this algorithm, features of both authentic and fake news documents are pulled out automatically. The researchers evaluated this approach on five datasets [5].

Z Khanam et al used feature selection and processing of attributes to apply machine learning models like SVM, Logistic regression, KNN, XGBoost and obtained the accuracy of 0.92 using SVM model on the dataset imported from Kaggle[6].

Veronica Perez-Rosas used different models, including the decision tree, naive Bayes algorithms and clustering were employed to differentiate between Twitter spam senders. On average, the models were able to identify spammers with an accuracy of 0.70 and fraudsters with an accuracy of 0.712. However, the models exhibited a low level of intermediate precision in distinguishing spammers from non-spam accounts[7].

Their study assumes that Fake news detection is an application of predictive analysis that involves three stages: cleaning/processing, feature extraction, and classification or categorising. A hybrid classification model designed specifically for identifying counterfeit news was used in this study. The classification approach involved a combination of random forests and KNN. Metrics like accuracy, precision and recall for the proposed models were calculated, and the results showed an improvement of up to 8 percent using a hybrid model for fake news detection[8].

Alim Al Ayub Ahmed and team in their research tried to identify which models can and can not be used for fake news classification and the steps to implement them[9]. Fahim Belal Mahmud and team applied models like SVM, LR, Random forest using the spacey and BERT techniques and using four GNN vaiants : GAT,GraphSAGE, GCN, and GIN.[10]

III. METHODOLOGY

This study utilized a three-stage approach for its methods. In the first stage, the raw dataset underwent pre-processing which involved filtering and various data cleaning techniques to extract semantic features. A stopword filter was implemented to remove prepositions and categorize the data. The second stage involved the conversion of text based features into binary vectors through numerical techniques. The final stage utilized both machine learning and deep learning algorithms like Naive Bayes, Random Forest, XGB Classifier to create partitions within the dataset. The same methodology is shown through figure 1.

A. The Pre-processing Stage

Data preprocessing involves the cleaning and transformation of unstructured data, such as text, to prepare it for analysis. Text mining is a common approach to this task, but it can be challenging due to the presence of impurities like HTML tags, single characters, numbers, apostrophes and unwanted symbols. In this study, we applied the stopwords removal method to clean and preprocess a classified dataset. Stopword removal is a widely used data filtering technique in information retrieval and text classification. It involves eliminating certain common words like: ”the, in, a, an, with”, that have little impact on the categorization of data.

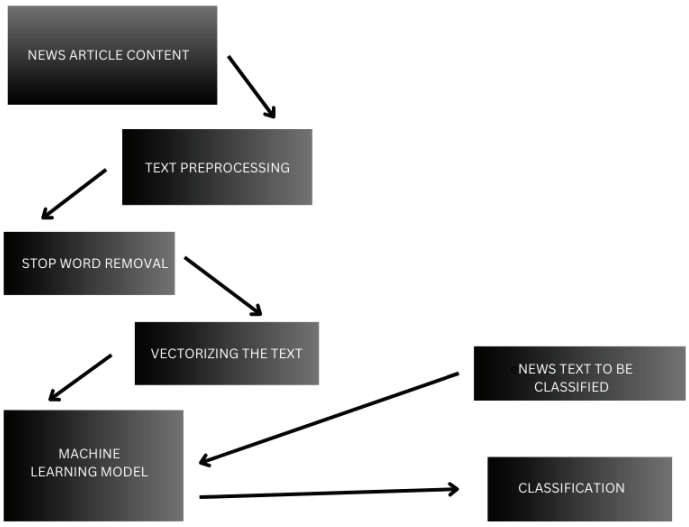

Fig. 1. Structure of this Paper

B. The Extract Features Stage

In this technique we use a method to convert the textual data to binary vectors 0 and 1 to make it convenient to train the classification algorithms. New vectors are created from each new entry in the sample text file. Vectors can be created using multiple techniques mentioned below:

1. TF-IDF vectorizer: The term frequency-inverse document frequency (TF-IDF) vectorizer is one of the most common method for the first stage i.e Feature Extraction. This technique involves two stages, with the first stage calculating the term frequency (TF), and the second stage computing the inverse document frequency (IDF).

Term Frequency = (Number of times the term appears in a document) / (Total number of terms in the document)

Inverse Document Frequency = log(Total number of documents / Number of documents with the term)

2. N-gram level vectorizer: An n-gram level vectorizer is a technique used in natural language processing (NLP) to convert text data into numerical vectors that can be used for machine learning. N-grams are contiguous sequences of n words from a given text, and the vectorizer creates a feature matrix with the frequency count of each n-gram in the text. This approach captures the contextual information of the text, making it useful for various NLP tasks like sentiment analysis and text classification. N-gram level vectorizers are customizable and can generate feature matrices with different levels of granularity.

3. Character level vectorizer: A character-level vectorizer is a technique used in natural language processing to convert text data into numerical vectors by considering each character in the text as a separate feature. This approach captures the underlying structure and patterns of the text at a granular level and is useful for tasks such as language modeling and text generation.

C. The Classifiers Stage

The following Machine learning and deep learning algorithms were used in this study: Gaussian Naive Bayes, Bernoulli Naive Bayes, Multilayer Perceptron, Random Forest Classifier, XGBoost Classifier.

- **Gaussian Naive Bayes** is a classification algorithm that use Bayes' theorem and assumes that features are independent and normally distributed. It estimates the mean and variance of the input features for each class to compute the probability of each class given the input features. Using these estimations, the system then computes the likelihood of each feature value given the class. It derives the joint probability of the input feature vector given the class by multiplying the likelihoods for each feature. Finally, given the input feature vector, it computes the posterior probability of each class and selects the class with the highest probability as the predicted class.
- **Bernoulli Naive Bayes** is based on the assumption that the features are binary or boolean values. It estimates the likelihood of each feature for each class to compute the probability of each class given the supplied features. Using these probabilities, the algorithm then computes the likelihood of the input feature vector given the class. It derives the joint probability of the input feature vector given the class by multiplying the likelihoods for each feature. Finally, given the input feature vector, it computes the posterior probability of each class and selects the class with the highest probability as the predicted class. Bernoulli Naive Bayes is a text classification method in which the features indicate the presence or absence of specific words in a document.
- **MLP (Multilayer Perceptron) Classifier** is an artificial neural network that is commonly utilized for supervised learning tasks such as classification. It comprises of several layers of interconnected nodes, where each node carries out a weighted sum of its inputs and applies an activation function to generate its output. MLPs are extensively used in various applications such as image recognition, speech recognition, and natural language processing. They have the ability to learn intricate non-linear connections between inputs and outputs and can be trained using the backpropagation algorithm.
- **Random Forest Classifier** is an ensemble learning technique that enhances the accuracy and consistency of predictions by combining multiple decision trees. It creates numerous decision trees during training and then determines the class by identifying the mode of the predicted classes by individual trees. Random Forests have the capability to handle large datasets with high dimensionality and learn non-linear relationships between input features and output classes. They are commonly used in fields such as finance, medicine, and e-commerce for various applications like fraud detection, disease diagnosis, and customer segmentation. Moreover, Random Forests provide feature importance scores that can be used for data interpretation and feature selection.
- **XGBoost (Extreme Gradient Boosting) Classifier** algorithm is a robust machine learning technique that is extensively used for supervised learning problems like classification and regression. It is a highly optimized implementation of the gradient boosting algorithm, which uses a combination of multiple decision trees to enhance the accuracy and efficiency of predictions. XGBoost applies a gradient descent algorithm to optimize the objective function and prune the decision trees to avoid overfitting. This algorithm can handle missing values and feature interactions, and it supports parallel processing to expedite model training. XGBoost is utilized in various fields such as finance, healthcare, and marketing for tasks like credit risk assessment, disease diagnosis, and customer churn prediction.

D. Results

The primary objective of this project is to perform a comprehensive analysis of the news data contained in the WELFake dataset, which is available on the Kaggle website. Our analysis involves the use of six distinct algorithms, namely:

- Bernoulli Naive Bayes
- MLPClassifier
- Gaussian Naive Bayes
- Random Forest Classifier
- XGBoost

These algorithms are chosen due to their proven effectiveness in the data driven field like data analysis and machine learning.

We conduct our analysis by first importing the dataset into the Anaconda platform and then executing the algorithm codes in Python. The cognitive learning library is used

to automatically generate a confusion matrix, which we use to visualize the results of our analysis. Additionally, performace was measured based on the calculated value of accuracy of each algorithm.

The confusion matrix and accuracy scores for each of the six algorithms used in the analysis are presented below. As can be seen, the XGBoost algorithm outperforms all the others with a high accuracy score of 94%, while the Gaussian Naive Bayes algorithm has the lowest accuracy score of 74%. The results obtained from our analysis provide valuable insights into the usefulness of different algorithms for detecting fake and deceptive news and can be used to inform future research in this area.

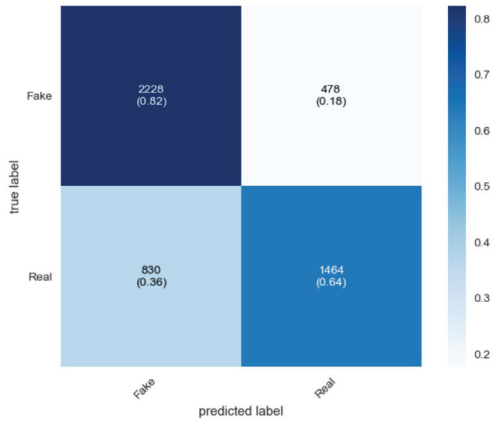

Fig. 2. Confusion Matrix for Gaussian Naive Bayes.

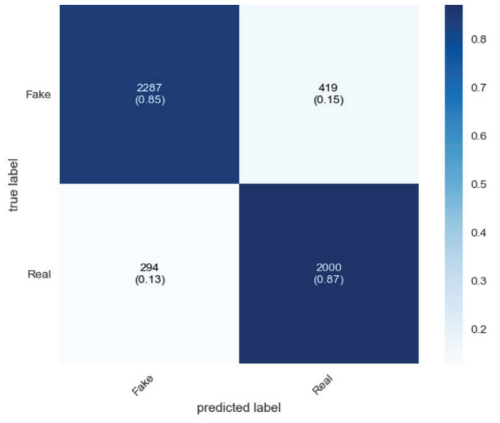

Fig. 3. Confusion Matrix for Bernoulli Naive Bayes.

CONCLUSION

The primary focus of this research paper is to address the growing issue of fake news on the internet, specifically

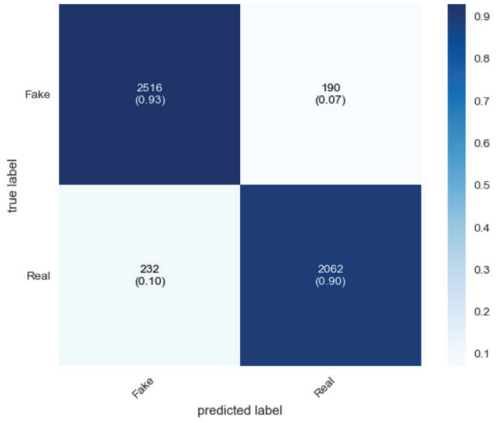

Fig. 4. Confusion Matrix for Multi Layer Perceptron.

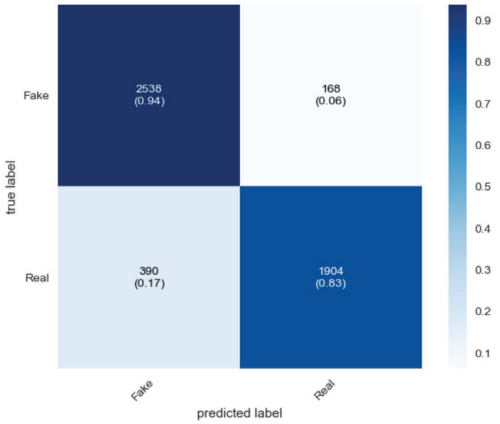

Fig. 5. Confusion Matrix for Random Forest Classifier.

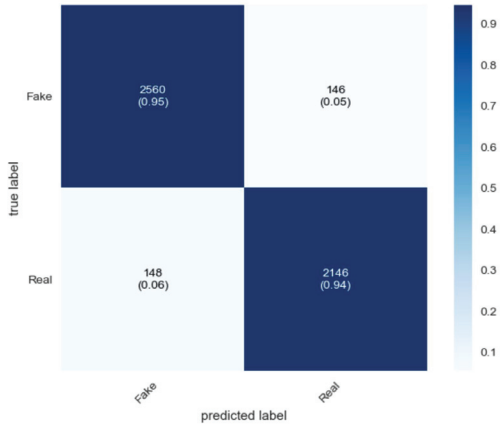

Fig. 6. Confusion Matrix for XG Boost Classifier.

TABLE I
RESULTS

Model	*Accuracy(Train)*	*Accuracy(Test)*	*CV Score Mean(Train)*
GaussianNB	1.00	0.74	0.72
BernoulliNB	0.91	0.86	0.85
MLPClassifier	1.00	0.92	0.90
RandomForest	1.00	0.89	0.89
XGBClassifier	1.00	0.94	0.95

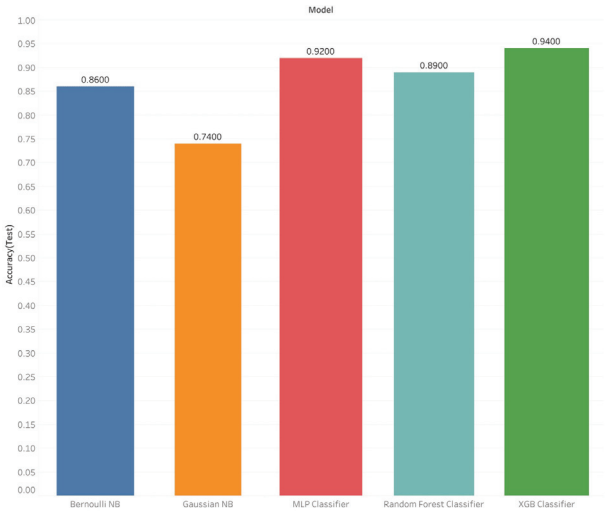

Fig. 7. Accuracy Scores

regarding textual content. The paper aims to provide a comprehensive analysis of previous studies and advancements in the field, and to conduct a comparative analysis on the WELFake dataset available on Kaggle.com. The main objective of the study is to determine the accuracy of various algorithms in classifying news as either real or fake. To achieve this, five different algorithms were applied to the dataset: GaussianNB, BernoulliNB, MLPClassifier, RandomForestClassifier, and XGBClassifier.

The findings of the study indicate that XGBClassifier exhibited the highest accuracy, while GaussianNB demonstrated the lowest accuracy among the tested algorithms. However, the study's results are limited due to the use of only one dataset, and there is significant potential for further research and development of new algorithms to expand the classification beyond textual content to non-textual media such as fake images or videos.

In conclusion, the study underscores the importance of developing effective algorithms to combat the proliferation of fake news on the internet. The use of advanced algorithms in conjunction with human oversight can significantly improve the accuracy and reliability of news classification, thereby reducing the impact of fake news on society.

REFERENCES

[1] Akshay Jain and AmeyKasbe. "Fake News Detection." 2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS). Bhopal, India: IEEE.2018.

[2] A. Farzana Islam, et al. "Effect of Corpora on Classification of Fake News using Naive Bayes Classifier." International Journal of Automation, Artificial Intelligence and Machine Learning 1.1 (2020): 80-92.

[3] P. Bharadwaj, and Z. Shao. "Fake news detection with semantic features and text mining." International Journal on Natural Language Computing (IJNLC) Vol 8 (2019).

[4] C. Carlos, M. Mendoza, and Barbara Poblete. "Information credibility on twitter." Proceedings of the 20th international conference on World wide web. 2011.

[5] K., Hamid, and Jiliang Tang. "Learning hierarchical discourse-level structure for fake news detection." arXiv preprint arXiv:1903.07389 (2019).

[6] Fake News Detection Using Machine Learning Approaches To cite this article: Z Khanam et al 2021 IOP Conf. Ser.: Mater. Sci. Eng. 1099 012040.

[7] Pérez-Rosas, V., Kleinberg, B., Lefevre, A., Mihalcea, R. (2017). Automatic detection of fake news. arXiv preprint arXiv:1708.07104.

[8] Looijenga, M. S. "The Detection of Fake Messages using Machine Learning." 29 Twente Student Conference on IT, Jun. 6th, 2018, Enschede, The Netherlands. Netherlands: y.utwente.nl. 2018.

[9] Ahmed, Alim Al Ayub, et al. "Detecting fake news using machine learning: A systematic literature review." arXiv preprint arXiv:2102.04458 (2021).

[10] Mahmud, Fahim Belal, et al. "A comparative analysis of Graph Neural Networks and commonly used machine learning algorithms on fake news detection." 2022 7th International Conference on Data Science and Machine Learning Applications (CDMA). IEEE, 2022.

[11] E. C. Tandoc Jr et al. "Defining fake news a typology of scholarly definitions". Digital Journalism, 1–17. 2017.

[12] Gilda, S. "Evaluating machine learning algorithms for fake news detection." 15th Student Conference on Research and Development (SCOReD) (pp. 110-115). IEEE. 2017.

[13] Supanya Aphiwongsophon et al. " Detecting Fake News with Machine Learning Method." 2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON). Chiang Rai, Thailand, Thailand: IEEE . 2018.

[14] Abdullah-All-Tanvir, Mahir, E. M., Akhter S., Huq, M. R. (2019). Detecting Fake News using Machine Learning and Deep Learning Algorithms. 7th International Conference on Smart Computing Communications (ICSCC), Sarawak, Malaysia, Malaysia, 2019, pp.1-5, https://doi.org/10.1109/ICSCC.2019.8843612

[15] Granik, M., Mesyura, V. (2017). Fake news detection using naive Bayes classifier. 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), Kiev, pp.900-903, https://doi.org/10.1109/UKRCON.2017.8100379

[16] S. Helmstetter, and H. Paulheim. "Weakly supervised learning for fake news detection on Twitter." 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM). IEEE, 2018.

[17] Roy, Arjun, et al. "A deep ensemble framework for fake news detection and classification." arXiv preprint arXiv:1811.04670 (2018).

[18] T. H. Borkar and T. Ahuja, "Comparative Study of Supervised Learning Algorithms for Fake News Classification," 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli,