**Comparative Analysis of Machine Learning Classifiers for**

**Fake News Detection**

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

**Bachelor of Technology**

In

**Computer Science and Engineering**

**School of Engineering and Sciences**

Submitted by

**Tarun Vaka AP21110011363**

**Jayasree Meda AP21110011375**

**Divya Sree Kovvuru AP21110011418**

**A picture containing text

Description automatically generated**

Under the Guidance of

**Dr. M Naveen Kumar**

**SRM University–AP**

**Neerukonda, Mangalagiri, Guntur**

**Andhra Pradesh – 522 240**

**May, 2024**

# **Certificate**

Date: 13-May-24

This is to certify that the work present in this Project entitled “Comparative Analysis of Machine Learning Classifiers for Fake News Detection” has been carried out by Divya Sree Kovvuru, Jayasree Meda, Tarun Vaka under my supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology in School of Engineering and Sciences.

Supervisor

(Signature)

Dr. M Naveen Kumar,

Assistant Professor,

Department of Computer Science and Engineering.

# **Acknowledgements**

We would like to express our sincere gratitude to all those who contributed to the successful completion of this research project on **"Automated Image Classification: Enhancing CI/CD with Deep Learning Models."**

First and foremost, We would like to convey our sincere gratitude to Dr. Sanjay Kumar, my advisor, for his essential advice, mentorship, and steadfast support during the course of my project. The direction and caliber of this work were greatly influenced by their knowledge and observations. My gratitude goes out to the members of the project team who worked together on different facets of this endeavor. Their combined knowledge and hard work substantially increased the breadth and depth of our conclusions.

We are very grateful to the contributors of open-source tools and datasets that were instrumental in the development and validation of our methodologies. Their shared resources facilitated the robustness and reproducibility of our research outcomes. Lastly, We would like to express our gratitude to our friends and family for their patience, understanding, and support throughout the challenging stages of this project.

This collaborative effort would not have been possible without the collective support and dedication of everyone involved, and for that, I am sincerely thankful.

# **Table of Contents**

| Certificate……………………………………………………………………………………….2  Acknowledgements……………………………………………………………………………..3  Table of Contents………………………………………………………….…………………….4  List of Figures…………………………………………………………………………………..5  Abbreviations…………………………………………………………………………………...6  Abstract…………………………………………………………………………………………7  Introduction……………………………………………………………………………………..8  Methodology…………………………………………………………………………………..10  Dataset Description……………………………………………………………………………14  Technologies used……………………………………………………………………………..15  Output………………………………………………………………………………………….16  Limitations……………………………………………………………………………………..17  Future work……………………………………………………………………………………18  Conclusion………………………………………………………………….………………….19  References………………………………………………………………….………………….20 |
| --- |
|  |

# 

# **List of Figures**

Figure 1. Output of Not Fake News Detection for given sample text…………………16

# **Abbreviations**

LG Linear Regression

RF Random Forest

DT Decision Tree

GB Gradient Boosting

# **Abstract**

The proliferation of fake news in today's information landscape necessitates effective automated detection methods. In this study, we compare four prominent machine learning classifiers: Logistic Regression, Random Forest, Decision Tree, and Gradient Boosting. We utilize a diverse dataset comprising textual features extracted from news articles, encompassing linguistic patterns, syntactic structures, and semantic cues. Through extensive experiments, we assess each classifier's ability to distinguish between genuine and fake news articles. Evaluation metrics such as accuracy, precision, recall, and F1 score are employed to gauge model performance. Findings reveal nuanced differences in classifier efficacy. Logistic Regression offers simplicity and speed, making it suitable for real-time applications. Random Forest exhibits robustness to noise and overfitting, especially beneficial for high-dimensional datasets. Decision Tree provides interpretability, facilitating insight into the decision-making process. Gradient Boosting, with its ensemble approach, demonstrates superior predictive performance by optimizing weak learners sequentially. Additionally, feature importance analysis identifies discriminative features contributing to classification. These insights offer valuable cues for understanding fake news propagation and detection mechanisms.The project utilizes the TF-IDF (Term Frequency-Inverse Document Frequency) representation to preprocess the textual data, capturing the importance of words within each document relative to the entire corpus. Through a comparative analysis of these classifiers, we aim to discern their effectiveness in detecting deceptive content and contribute to the development of robust fake news detection systems. Overall, this study sheds light on classifier strengths and limitations in fake news detection, informing the development of more effective detection systems.

# **Introduction**

The rise of fake news in today's digital era poses a significant challenge to the integrity of information dissemination and public discourse. With the rapid expansion of online platforms and social media networks, misinformation can spread virally, influencing public opinions, shaping perceptions, and even impacting critical decision-making processes. Recognizing the gravity of this issue, researchers and practitioners have turned to machine learning techniques to develop automated systems capable of detecting fake news articles. These systems aim to analyze textual content, identify deceptive patterns, and classify articles as either genuine or fake with a high degree of accuracy.

In the pursuit of constructing a robust fake news detection system, a thorough exploration of various machine learning models revered for their prowess in classification tasks was undertaken. The repertoire of models encompassed a diverse array of algorithms, each offering unique strengths and capabilities. An ensemble comprising Logistic Regression, Decision Tree, Gradient Boosting, and Random Forest was strategically selected to leverage their distinct advantages in discerning patterns and making predictions within complex datasets. Central to the approach was the utilization of TF-IDF (Term Frequency-Inverse Document Frequency), a powerful technique in natural language processing, to preprocess the textual data. TF-IDF captures the significance of terms within individual documents by weighing them based on their frequency in the document and inversely proportional to their frequency across the entire corpus. This preprocessing step ensured that the machine learning models could effectively extract meaningful features from the textual data, enhancing their ability to differentiate between genuine and fake news articles.

The investigation began with a comprehensive exploration of the TF-IDF representation, analyzing its impact on feature extraction and model performance. Delving into the intricacies of term frequency and inverse document frequency elucidated how they contributed to the representation of textual data and influenced the decision-making process of the classifiers. Subsequently, the selected machine learning models were implemented and trained on the preprocessed data. Logistic Regression, known for its simplicity and interpretability, served as a baseline model, providing insights into the linear relationship between features and the target variable. Decision Tree, characterized by its hierarchical structure and intuitive decision-making process, offered transparency and ease of understanding in classifying news articles as genuine or fake. Meanwhile, Random Forest, a powerful ensemble method, harnessed the collective intelligence of multiple decision trees to improve predictive accuracy and robustness against overfitting. Finally, Gradient Boosting, through iterative optimization of weak learners, strived to minimize prediction errors and achieve superior performance in fake news detection.

Throughout the experimentation phase, meticulous evaluation of the performance of each classifier was conducted using established metrics such as accuracy, precision, recall, and F1 score. Hyperparameters were fine-tuned, cross-validation was conducted, and ensemble techniques were explored to optimize model performance and ensure generalizability to unseen data. The analysis extended beyond mere performance metrics as a delve into the interpretability of the models was performed, investigating feature importance and decision boundaries to gain insights into the underlying mechanisms of fake news detection. Through a systematic comparison of the classifiers, the aim was to identify the most effective model for detecting fake news while elucidating the strengths and limitations of each approach. Ultimately, the endeavor sought to contribute to the ongoing efforts to combat misinformation and uphold the integrity of information dissemination in the digital age.

# 

# 

# **Methodology**

## **Importing Libraries**

* The pandas library is crucial for data manipulation, as it provides data structures and functions to efficiently handle tabular data.
* NumPy is essential for numerical operations and array manipulation, which are fundamental for processing data in machine learning.
* Seaborn and Matplotlib.pyplot are used for data visualization, enabling the creation of informative plots and charts to gain insights from the data.
* From sklearn.model\_selection, train\_test\_split is imported to split the dataset into training and testing sets, facilitating model evaluation.
* The accuracy\_score and classification\_report functions from sklearn.metrics are imported to evaluate the performance of the trained models.

## **Loading Data**

* The Fake.csv and True.csv datasets are read into pandas DataFrames using the read\_csv() function, allowing easy access and manipulation of the data.
* A binary classification label is added to each DataFrame to distinguish between fake (0) and true (1) news articles, facilitating supervised learning.
* Manual testing datasets are created by selecting the last 10 rows from each DataFrame, providing a separate dataset for manual validation of model predictions.

## **Preprocessing Data**

* To ensure data quality and consistency, irrelevant columns such as title, subject, and date are dropped using the drop() function.
* Missing values are checked using the isnull() function, ensuring that the dataset is clean and complete before further processing.
* The DataFrame is shuffled using the sample() function with frac=1 parameter to randomize the order of rows, preventing bias during model training.

## **Text Processing Function**

* The wordopt() function is defined to preprocess the text data, ensuring uniformity and consistency in the textual features.
* This function applies a series of text processing techniques such as converting text to lowercase, removing special characters, URLs, punctuation, HTML tags, digits, and extra whitespaces using regular expressions and string manipulation.

## **Defining Dependent and Independent Variables**

* The 'text' column is designated as the independent variable (x), representing the textual content of the news articles.
* The 'class' column is assigned as the dependent variable (y), indicating whether the news articles are fake (0) or true (1), facilitating supervised learning.

## **Splitting Training and Testing Data**

* The train\_test\_split() function from sklearn.model\_selection is employed to split the dataset into training and testing sets.
* By specifying the test\_size parameter (typically set to 0.25), 75% of the data is allocated for training, and 25% for testing, ensuring a sufficient amount of data for model evaluation.

## **Converting Text to Vectors**

* The TfidfVectorizer from sklearn.feature\_extraction.text is utilized to convert the textual data into numerical vectors.
* This vectorization process transforms the text data into a matrix of TF-IDF features, capturing the importance of words in each document relative to the entire corpus, thereby facilitating machine learning model training.

## **Model Training**

* Logistic Regression (LR), Decision Tree (DT), Gradient Boosting Classifier (GBC), and Random Forest Classifier (RFC) models are trained on the training data (x\_train, y\_train).
* The fit() method is applied to each model to learn patterns and relationships between the features (TF-IDF vectors) and the target variable (fake or true news).

## **Model Testing**

* The performance of each trained model is evaluated on the testing data (x\_test, y\_test) to assess their predictive capabilities.
* The predict() method is used to generate predictions for the test data, and the accuracy\_score and classification\_report functions are employed to quantify the performance of each model.

## **Manual Testing Function**

* A manual\_testing() function is defined to simulate manual testing of the trained models using custom news articles provided by the user.
* This function preprocesses the input text, converts it into TF-IDF vectors using the trained TfidfVectorizer, and generates predictions using the trained models (LR, DT, GBC, RFC).

## **Input Text and Detect Fake News or Not**

* The user is prompted to input a news article, which is then passed to the manual\_testing() function to predict whether it is fake or true using each of the trained models.
* The predictions are displayed to the user for interpretation, enabling them to assess the credibility of the news article based on the model predictions.

# **Dataset Description**

For our fake news detection project, we utilized two primary datasets stored in CSV format, providing a diverse collection of textual data for training and evaluating our machine learning models. The first dataset, labeled "true.csv," contained news articles verified to be accurate and truthfully reported events. Each entry in this dataset included four key attributes: title, text, subject, and date. The title and text fields contained the headline and content of the news article, respectively, while the subject field categorized the news into specific topics. The date field indicated the publication date of each news article.

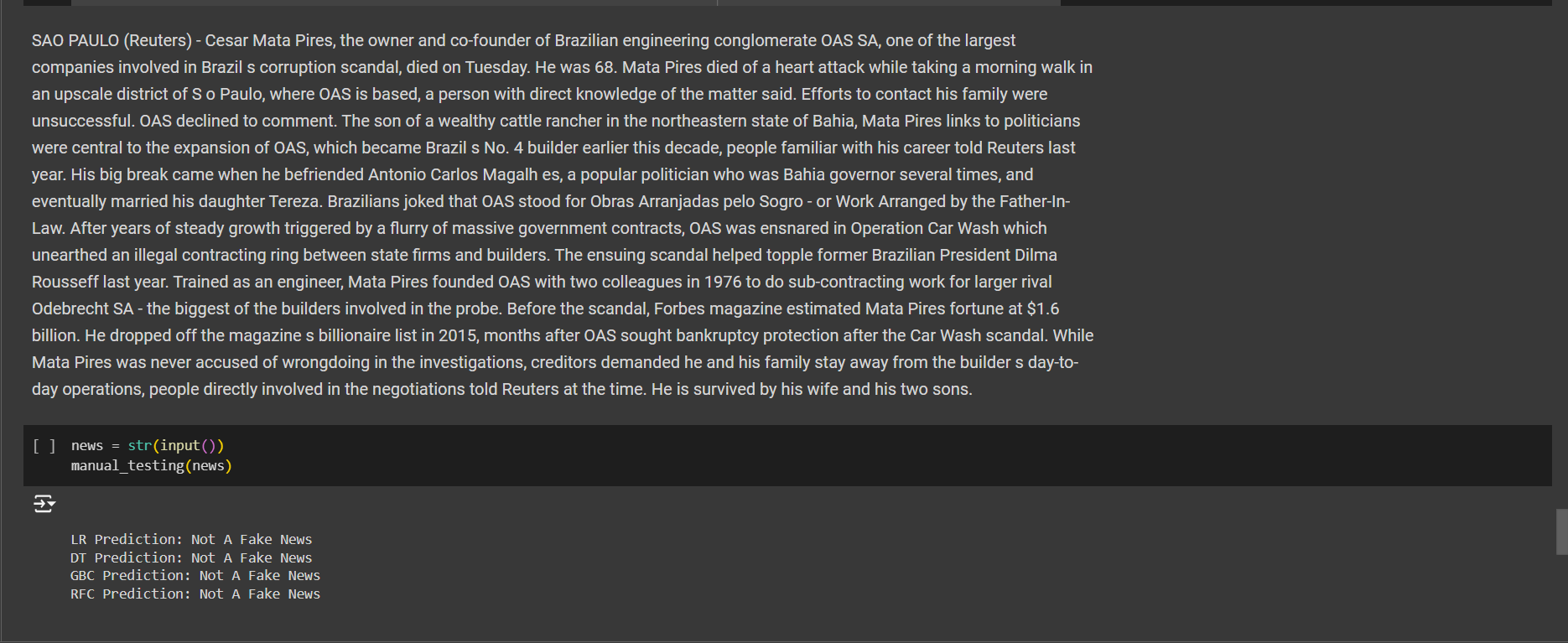
Conversely, the second dataset, labeled "fake.csv," consisted of news articles deliberately fabricated or misleading. Like the "true.csv" dataset, each entry in this dataset also contained four attributes: title, text, subject, and date. However, the content of these articles was intentionally deceptive, designed to misinform or manipulate readers. The "fake.csv" dataset served as a crucial component for training our machine learning models to distinguish between genuine and deceptive news content. The "true.csv" dataset had a size of 21,417 entries and 5 columns, while the "fake.csv" dataset had a shape of 23,481 entries and 5 columns. These datasets provided a comprehensive foundation for our fake news detection project, enabling us to develop and evaluate effective algorithms for identifying misinformation.

# **Technologies Used**

## **Software**

* **Python Programming Language:** The primary language used for data manipulation, modeling, and evaluation due to its extensive libraries and frameworks for machine learning, such as pandas, NumPy, scikit-learn, and TensorFlow.
* **Jupyter Notebooks or Integrated Development Environments (IDEs):** Provide interactive computing environments for prototyping, experimentation, and collaboration on machine learning projects. Jupyter Notebooks offer a convenient interface for writing code, visualizing results, and documenting analyses.
* **Pandas:** Used for data manipulation and analysis, offering powerful data structures and functions for cleaning, transforming, and exploring tabular data.
* **NumPy:** Essential for numerical computing and array manipulation, providing efficient data structures and mathematical functions for handling large datasets.
* **Scikit-learn:** A comprehensive machine learning library in Python, offering tools for classification, regression, clustering, dimensionality reduction, and model evaluation.
* **Matplotlib and Seaborn:** Visualization libraries used for creating informative plots, charts, and graphs to analyze data distributions, trends, and model performance.
* **Text Editors:** Used for writing and editing code, scripts, and configuration files. Popular text editors include Visual Studio Code, Sublime Text, Atom, and PyCharm.
* **Git and Version Control Systems:** Employed for managing code repositories, tracking changes, and collaborating with team members on machine learning projects. Version control systems enable developers to maintain a history of code revisions, rollback changes, and merge contributions seamlessly.

# **Output**



**Figure 1. Output of Not Fake News Detection for given sample text**

# **Limitations**

Limitations persist within the domain of fake news detection systems, posing significant challenges to their effectiveness and practicality. One key constraint lies in the availability and quality of training data. Often, datasets containing labeled instances of both fake and genuine news articles are limited in size and may not adequately represent the diverse spectrum of misinformation encountered in real-world scenarios. This scarcity of data undermines the ability of machine learning models to generalize effectively, leading to suboptimal performance when confronted with novel instances of fake news. Moreover, inherent biases within the training data further complicate matters. These biases can stem from various sources, including the selection criteria for news sources, subjective labeling processes, or imbalances between fake and authentic news samples. Such biases distort the learned patterns, undermining the model's capacity to accurately differentiate between legitimate and deceptive content.

Furthermore, the feature representation methods employed in fake news detection systems may present limitations. Traditional techniques such as TF-IDF vectorization may struggle to capture the intricate linguistic nuances and contextual cues inherent in news articles. Consequently, important information necessary for accurate classification may be lost in the process, hindering the model's ability to discern subtle indicators of fake news. Additionally, while sophisticated machine learning algorithms may achieve high levels of accuracy, their complexity often comes at the expense of interpretability. The opaque nature of these algorithms makes it challenging for users to comprehend and trust the decision-making processes underlying the system's outputs. Ethical considerations, such as the risk of censorship, privacy infringements, and the inadvertent amplification of biases, further underscore the need for responsible development and deployment of fake news detection technologies to mitigate potential societal harms.

# **Future Work**

Future work in the field of fake news detection presents numerous avenues for exploration and refinement. One promising direction involves leveraging advanced natural language processing (NLP) techniques to enhance the semantic understanding of news articles. Deep learning architectures such as transformers and pre-trained language models like BERT, GPT, and XLNet offer powerful tools for capturing complex linguistic patterns and contextual relationships within text data. By integrating these models into fake news detection systems, researchers can potentially improve the accuracy and robustness of classification algorithms, enabling more nuanced analysis of news content and better identification of deceptive narratives. Additionally, incorporating multimodal approaches that integrate textual, visual, and metadata features could further enhance the effectiveness of fake news detection systems. By considering multiple modalities simultaneously, these systems can capture a more comprehensive view of news articles, potentially uncovering subtle cues and inconsistencies that may indicate the presence of misinformation.

Moreover, future research efforts may focus on developing more transparent and interpretable machine learning models for fake news detection. Explainable artificial intelligence (XAI) techniques, such as attention mechanisms, saliency maps, and feature importance visualization, can provide insights into the decision-making processes of complex models. By enhancing interpretability, these methods enable users to understand how models arrive at their predictions, fostering trust and facilitating human-machine collaboration in the identification of fake news. Additionally, exploring the integration of domain-specific knowledge and expert insights into fake news detection systems could yield valuable improvements. By leveraging domain expertise from journalists, fact-checkers, and subject matter experts, these systems can incorporate contextual understanding and domain-specific heuristics, enhancing their ability to discern between credible and deceptive news content effectively.

# **Conclusion**

In conclusion, the pursuit of mitigating the impact of fake news through machine learning methodologies has revealed both progress and ongoing challenges. Despite encountering obstacles such as data scarcity, biases, and interpretability issues, significant advancements have been achieved. Among the various models explored in this project, it's notable that the decision tree classifier emerged as particularly promising, demonstrating its efficacy in discerning between fake and authentic news articles. This finding underscores the importance of exploring diverse algorithms and methodologies, as it showcases how a seemingly straightforward approach can yield remarkable results. Moving forward, integrating advanced natural language processing techniques, embracing multimodal approaches, and ensuring transparency in model decision-making processes will be pivotal in further refining fake news detection systems. Collaboration across disciplines and the continued integration of ethical considerations will be paramount as we strive to build more resilient and effective solutions to combat the proliferation of misinformation in our digital landscape.

# 

# **References**

1. Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H. (2019). FakeNewsNet: A Data Repository with News Content, Social Context and Dynamic Information for Studying Fake News on Social Media. arXiv preprint arXiv:1906.01226.
2. Ruchansky, N., Seo, S., & Liu, Y. (2017). CSI: A hybrid deep model for fake news detection. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (pp. 797-806).
3. <https://www.geeksforgeeks.org/fake-news-detection-using-machine-learning/>
4. <https://www.simplilearn.com/tutorials/machine-learning-tutorial/how-to-create-a-fake-news-detection-system>
5. <https://arxiv.org/pdf/2102.04458/1000>