International Journal of Computer and Information System (IJCIS)
Peer Reviewed – International Journal
Vol : Vol. 06, Issue 03, August 2025
e-ISSN : 2745-9659
https://ijcis.net/index.php/ijcis/index

Evaluating Machine Learning Algorithms for Detecting Online Text-based Fake News Content

Deni Kurnianto Nugroho[1] , Marwan Noor Fauzy[2], Kardilah Rohmat Hidayat[3]
Department of Information System, Faculty of Computer Science
Universitas Amikom Yogyakarta
Yogyakarta, Indonesia
deni@amikom.ac.id[1] , marwannoorfauzy@amikom.ac.id[2], kardilah.rh@amikom.ac.id[3]

Abstract*—The rapid spread of disinformation and fabricated news across online platforms poses a critical risk to informed public engagement and the foundations of democratic governance. This study examines how well different machine learning techniques can classify fake news, using textual features extracted through the Term Frequency–Inverse Document Frequency (TF-IDF) method. The analysis includes five commonly used algorithms like Logistic Regression, Support Vector Machine (SVM), Naive Bayes, Random Forest, and XGBoost. A publicly accessible dataset containing annotated real and fake news articles served as the basis for training and testing these models. Dataset underwent extensive preprocessing, including tokenization, stopword removal, and TF-IDF vectorization, resulting in a sparse high-dimensional matrix of 5068 documents and 39,978 features. Performance evaluation was based on multiple metrics: train/test accuracy, misclassification rate, false positives/negatives, cross-validation mean score, and execution time. Results showed that SVM and Logistic Regression achieved the highest test accuracy (93.61% and 92.27%, respectively) and exhibited robust cross-validation scores, indicating strong generalization ability. In contrast, Naive Bayes produced faster results but suffered from a high false positive rate and lower accuracy (84.77%). Random Forest and XGBoost demonstrated good predictive power but showed signs of overfitting and moderate misclassification rates. These findings suggest that SVM and Logistic Regression are well-suited for fake news detection in textual datasets using TF-IDF features. While traditional models remain effective, future work may explore deep learning approaches and context-aware language models to enhance detection accuracy across more complex and multilingual datasets. This study contributes to the ongoing efforts to combat misinformation through automated, scalable, and interpretable machine learning techniques.*

Keywords *: fake news, text classification, tf-idf, machine learning, supervised learning*

I. INTRODUCTION

The development of information technology has brought significant changes in the way people access and disseminate information. One consequence of this ease of use is the increasing spread of false information or fake news, particularly through online media platforms and social media [1]. This phenomenon not only disrupts the democratic process, as seen in the 2016 US elections [2], but also has the potential to cause social unrest, misinformation on health issues, and division within society [3].

Manually identifying fake news has become increasingly impractical due to the massive scale of online information and the speed at which it spreads. As a result, Machine Learning (ML) techniques have gained traction as an effective solution to this issue [4][5]. These techniques allow systems to learn from data and detect patterns that help distinguish between authentic and fabricated news based on textual features [6]. ML-based methods have shown promising results across a range of tasks, particularly in the domains of natural language processing (NLP) and automated text classification [7].

Several commonly adopted algorithms for fake news detection include Support Vector Machine (SVM), Random Forest (RF), Logistic Regression (LR), and Naive Bayes (NB) [8]. The effectiveness of each algorithm often depends on factors such as the chosen text representation method (e.g., TF-IDF or word embeddings), the nature of the features, and the size and quality of the dataset. Prior studies have also highlighted that no single algorithm universally outperforms others in all scenarios [9], emphasizing the importance of comparative evaluation to determine the most suitable model for specific use cases.

In the context of online news, other challenges include variations in writing styles, biases in news sources, and the presence of clickbait and disguised opinions that are difficult to distinguish from fact [10]. Therefore, an effective classification approach must be able to capture the linguistic and semantic nuances of news text.

This paper investigates how different machine learning classifiers perform in the task of identifying fake news articles, using textual features derived from a well-known benchmark dataset. The analysis places emphasis on standard evaluation criteria including accuracy, precision, and recall while also incorporating assessments of model consistency through five-fold cross-validation and the time required for training and prediction.

Among the models tested, Logistic Regression and Support Vector Machine (SVM) consistently demonstrate robust results, particularly with regard to accuracy and recall, and exhibit stable performance across validation folds. Naive Bayes offers an appealing balance of speed and accuracy, making it a practical option for real-time or resource-constrained scenarios. In comparison, ensemble approaches such as Random Forest and XGBoost achieve strong predictive outcomes, albeit with longer processing times. These results illustrate the balance practitioners must strike between accuracy, interpretability, and computational cost when developing fake news detection systems.

Beyond technical challenges, fake news detection also has social and ethical dimensions that cannot be ignored. The algorithm used must be able to handle data bias, as well as consider the implications of misclassification, such as risks to freedom of expression or the reputation of the party being misreported [11][12]. Therefore, the development of an accurate and responsible fake news detection system requires a multidisciplinary approach, encompassing technical, social, and policy aspects. This research seeks to support this direction by providing a quantitative analysis of the performance of various machine learning algorithms in the context of text-based fake news detection, as a basis for the development of more reliable and ethical systems in the future.

II. RESEARCH METHODS

This study seeks to construct an automated system for identifying fake news through the application of machine learning techniques. The primary goal of the classification model is to effectively differentiate between authentic (REAL) and deceptive (FAKE) news content using the textual information provided. The methodological framework encompasses several key stages: exploratory data analysis, text cleaning and preprocessing, transformation of text into numerical features, model training, performance assessment, and examination of influential features that drive the model's predictions.

2.1 Dataset Description

The dataset employed in this study originates from a publicly available corpus comprising 6335 news articles with assigned labels. The class distribution is nearly even, consisting of 3171 articles identified as genuine (REAL) and 3164 labeled as false (FAKE). Each data point includes three primary components: the headline (title of the article), the main body of text (full news content), and the classification label indicating its authenticity. On average, each article contains approximately 776 words, offering sufficient textual depth for comprehensive content analysis.

2.2 Data Preprocessing

The goal of text preprocessing is to enhance the quality of textual data representation prior to model input. This process typically involves several steps: transforming all text to lowercase, eliminating punctuation and non-letter characters, removing frequently occurring stopwords, applying stemming using the Porter algorithm, and performing word tokenization. Once the text is cleaned and standardized, it is converted into a numerical vector using the TF-IDF technique. TF-IDF accounts for both the frequency of a word within a single document and its occurrence across the entire dataset [13], enabling the classification model to identify patterns by assigning greater importance to more informative terms.

2.3 Classification Algorithm

This study tested five machine learning algorithms for the purpose of fake news classification:

1) Logistic Regression
Linear models are often used for binary classification because of their ability to handle high-dimensional data efficiently [14].

2) Support Vector Machine (SVM)
Applied with a linear kernel to maximize the inter-class margin in the TF-IDF feature space. SVM is known to be robust to high dimensions and is suitable for text classification [15].

3) Naive Bayes
It is a simple and very efficient probabilistic model for text, because it assumes independence between features [16].

4) Random Forest
It is a decision tree-based ensemble algorithm that builds multiple models and aggregates the results to improve accuracy and reduce overfitting [17].

5) Extreme Gradient Boosting (XGBoost)
It is one of the modern boosting methods that iteratively optimizes the errors of the previous model. XGBoost is known for its high performance in various data mining competitions [18].

Each model was trained on a TF-IDF representation of the corpus, with standard hyperparameter settings to ensure fairness in performance comparisons..

2.4 Model Evaluation

To evaluate the classification performance, multiple complementary metrics were utilized, including training and testing accuracy, misclassification count, false positives, false negatives, and misclassification rate.

Accuracy measures the proportion of correct predictions out of all predictions, giving a general sense of overall performance. False positives (FP) indicate real news articles incorrectly classified as fake, which is critical to minimize to avoid mislabeling credible information. False negatives (FN) represent fake news articles incorrectly classified as real, which is equally important to reduce as they pose a risk of misinformation spread. Misclassification rate complements accuracy by showing the proportion of incorrect predictions.

These metrics collectively provide a more nuanced understanding of each model's strengths and weaknesses, especially in balancing the trade-off between FP and FN in fake news detection, where both types of errors carry significant societal consequences.

In addition to static accuracy values, 5-fold cross-validation was conducted to measure the generalizability of each model across different data splits. The mean cross-validation score (CV Mean) reflects average performance on unseen folds, while 2× standard deviation (CV 2×StdDev) offers insight into result variability and model stability [19]. This approach ensures that the evaluation is not biased toward a single train-test split and better represents real-

world performance.

The misclassification analysis focuses on two critical types of errors: false positives (misidentifying fake news as real) and false negatives (failing to detect fake news), which are essential in evaluating model reliability in real-world applications. Execution time for training and testing each model was also recorded to assess computational efficiency, particularly relevant when deploying models at scale.

For models that operate on linear decision boundaries, such as Logistic Regression and SVM, further analysis was performed on learned feature weights to identify influential words contributing to the classification of FAKE and REAL news. This interpretability aspect supports transparency and model explainability in sensitive domains like misinformation detection.

III. RESULT AND ANALYSIS

This section evaluates the performance of various machine learning approaches commonly used in text classification tasks. The models examined include linear classifiers such as Logistic Regression and SVM, probabilistic methods like Naive Bayes (NB), as well as ensemble-based techniques including Random Forest (RF) and XGBoost (XGB). These models are applied to the task of fake news detection using TF-IDF for textual feature representation. The evaluation considers key performance metrics accuracy, precision, recall, F1-score, and confusion matrix and employs 5-fold cross-validation to assess generalization capabilities. Further analysis of runtime and misclassification trends is conducted to identify model limitations. For interpretable algorithms, feature importance is also analyzed to uncover the most influential terms contributing to classification outcomes.

3.1 Exploratory Data Analysis (EDA)

Before training the classification model, an exploratory analysis of the data structure was conducted to understand the characteristics of the news content, particularly the number of words in the title and the article text. This is crucial for gaining initial insight into potential pattern differences between genuine and fake news.

1) Article Title Length Distribution

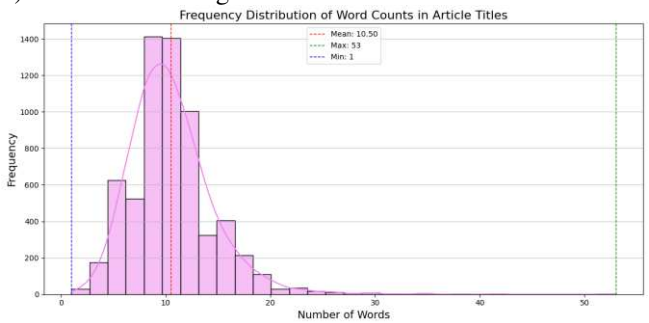

Figure 1. Frequency distribution of word counts in news article title

Figure 1 shows the frequency distribution of the number of words in article titles. The average title length is approximately 10.5 words, with a maximum length of 53 words and a minimum of just one word. The distribution shows a relatively normal pattern with a slight skew to the right. The majority of articles have titles between 7 and 14 words. The length of an article's title can be an early indicator in detecting fake news. Titles that are too short or too long can indicate an attempt to grab readers' attention, a common characteristic of clickbait content, often associated with fake news.

2) Distribution of Article Content Length

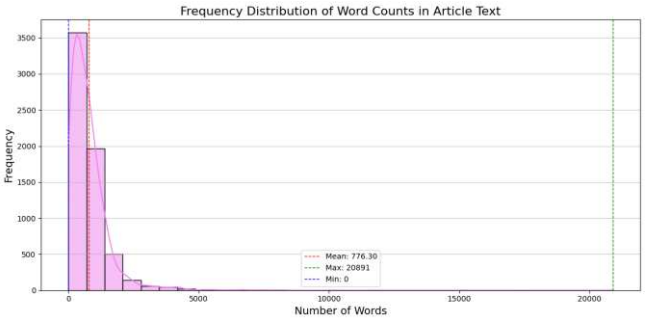

Figure 2. Frequency distribution of word counts in news article text

Figure 2 shows the word count distribution of the articles. The average article length is approximately 776 words, with the shortest article having 0 words and the longest exceeding 20,000 words. While there are extreme outliers, most articles are concentrated below 1,500 words.

This distribution indicates that the news content in the dataset is relatively varied in length. Extreme article lengths can impact the model's performance in feature extraction and should be taken into account during preprocessing or outlier trimming.

3) Comparison of Article Length by Label

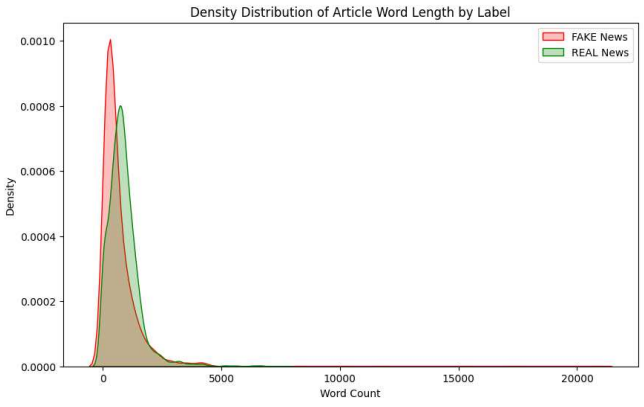

Figure 3. Density distribution of news article word length by label

Figure 3 displays the density distribution of article lengths by label (FAKE and REAL). It can be seen that articles labeled FAKE have a peak distribution at a lower word count compared to REAL articles. This indicates that fake news tends to be shorter and more to the point. This difference is consistent with previous research findings that suggest fake news is often structured in a more concise and emotional style to increase its spread on social media. Conversely, genuine news tends to be longer and includes detailed information and valid citations. While article length is not the sole factor in determining news authenticity, it can be used as an additional feature in the classification process, especially when used in conjunction with other semantic and syntactic features.

3.2 Preprocessing Stages

Before training a fake news detection model, the text data undergoes a series of preprocessing steps to ensure the quality and consistency of its representation. This step is crucial to ensure the machine learning model can capture relevant patterns and avoid distractions from noisy or redundant information.

The preprocessing process begins by converting all text to lowercase. This aims to equalize the representation of semantically similar but letter-different words, such as "Berita" and "berita." Next, the text is cleaned, removing special characters such as numbers, punctuation, symbols, and URLs that do not contribute to the core meaning of the news content. The cleaned text is then separated into tokens, or individual word fragments, through a tokenization process. To reduce the data dimensionality and reduce interference from overly common words, stopwords are also removed, such as "yang," "dan," "dalam," and so on. These stopwords do not provide significant information in the context of fake news classification because they occur frequently in almost all documents.

Although some NLP approaches also employ stemming or lemmatization to convert words to their base form, this implementation did not employ this process. This is because the text corpus used was already sufficiently clean and representative, and retaining the original word forms was considered to preserve a richer semantic context. The preprocessed textual data was transformed into numerical form using the TF-IDF method. This approach assigns weights to words based on how often they occur within an individual document compared to their occurrence across the full collection of documents. Terms that are common within a specific article but uncommon throughout the broader corpus are given higher importance, reflecting their relevance to that particular document.

The final result of this stage is a TF-IDF matrix with dimensions (5068, 39978), meaning there are 5,068 news articles and 39,978 unique word features resulting from preprocessing. This large feature size reflects the diversity of vocabulary in the news corpus and poses a challenge in model training due to the high risk of overfitting if not handled properly.

3.3 Classification Model Evaluation

After completing the training process using multiple classification algorithms, the models' ability to distinguish between real and fake news articles was carefully evaluated. The assessment involved key performance indicators such as accuracy, precision, recall, and F1-score, calculated on a test set containing 1,267 samples. To further ensure the reliability and generalizability of the results, a 5-fold cross-validation approach was implemented.

The evaluation outcomes are summarized in Table 1, presenting a comparative analysis based on commonly used classification metrics. Accuracy measures the overall correctness of predictions, precision captures how many of the predicted fake news items were actually fake, and recall evaluates the proportion of true fake news items that were successfully identified. All metrics were reported using macro-averaging to account for class imbalance and provide a balanced view of model performance.

The models were further evaluated through 5-fold cross-validation to assess their generalization performance. The average cross-validation accuracy and the corresponding ±2 standard deviations (CV ± 2 Std) are reported to reflect the stability and consistency of each model across different data splits. Furthermore, the total runtime of each model training and evaluation was included to provide insight into computational efficiency. This comprehensive evaluation framework helps in balancing predictive performance, reliability, and practical considerations, especially for critical tasks such as fake news detection.

In addition to classification metrics, the table includes cross-validation results to assess the consistency of model performance. The CV Mean refers to the average accuracy obtained across five folds, while 2×STD represents twice the standard deviation, capturing performance variability and model robustness. A lower standard deviation implies more stable and reliable predictions across different data splits. Furthermore, the table reports each model's training runtime, offering insights into computational efficiency, a critical factor for real-world deployment. By presenting these multiple dimensions of evaluation, Table 1 facilitates a well-rounded comparison of the models' predictive capabilities and practical considerations. Complementing this, Table 2 provides precision, recall, and F1-score for both FAKE and REAL classes, enabling a deeper understanding of class-specific strengths and weaknesses. This dual perspective ensures that model selection accounts not only for overall performance but also for the ability to handle the asymmetric costs of misclassifications in fake news detection.

Table 1. Benchmarking Classification Models for Fake News Identification

Model	Accuracy (Train/Test)	Miss-classified	False Positive	False Negative	Miss-classification Rate	CV Mean Score	CV StdDev x2	Execution Time (s)
LR	0.95/0.92	98	34	64	0.077	0.907	0.008	2.16
SVM	0.98/0.93	81	38	43	0.063	0.930	0.020	208.9
Naive Bayes	0.89/0.84	193	184	9	0.152	0.834	0.024	0.05
RF	1.0/0.91	110	50	60	0.086	0.086	0.010	9.71
XGBoost	0.99/0.92	101	47	54	0.079	0.914	0.013	56.64

Table 2. Precision, Recall, and F1-Score for Each Model

Model	Precision (Fake/Real)	Recall (Fake/Real)	F1-score (Fake/Real)	Macro Avg Precision	Macro Avg Recall	Macro Avg F1-score
LR	0.90/0.94	0.95/0.90	0.92/0.92	0.92	0.92	0.92
SVM	0.93/0.94	0.94/0.93	0.94/0.94	0.94	0.94	0.94
Naive Bayes	0.98/0.77	0.71/0.99	0.82/0.87	0.88	0.85	0.84
RF	0.91/0.92	0.92/0.91	0.91/0.91	0.91	0.91	0.91
XGBoost	0.92/0.93	0.93/0.91	0.92/0.92	0.92	0.92	0.92

1) Logistic Regression (LR)

The Logistic Regression model achieved a training accuracy of 95.17% and a testing accuracy of 92.27%, demonstrating strong generalization capabilities. It misclassified 98 out of 1,268 test samples, corresponding to a misclassification rate of 7.73%. The false positive count was 58, and false negatives were 40, indicating a relatively balanced misclassification pattern. From a cross-validation perspective, Logistic Regression had a mean CV score of 90.77%, with a 2× standard deviation of 0.83%, suggesting stable performance across folds. It also had a fast execution time of 2.16 seconds, making it a practical choice for large-scale applications.

Logistic Regression achieved balanced performance between precision and recall for both FAKE and REAL classes, with macro averages of 0.92 across all three metrics. The recall for FAKE news (0.95) was slightly higher than precision (0.90), indicating the model was particularly effective in capturing fake news articles, albeit at the cost of a slightly higher false positive rate. The F1-scores for both classes were identical (0.92), highlighting consistent predictive ability across categories. This balanced profile makes Logistic Regression a robust choice where both classes are equally important.

2) Support Vector Machine (SVM)

SVM delivered the highest test accuracy of all models, at 93.61%, and an impressive training accuracy of 98.58%. It had the lowest misclassification count (81 samples), equivalent to a misclassification rate of 6.39%. In terms of misclassification types, it produced 61 false positives and 20 false negatives, showing high sensitivity in detecting fake news. The model also had a high CV mean score of 93.07%, although its 2× standard deviation (2.06%) was slightly higher than that of Logistic Regression, indicating more variability across folds. However, the trade-off was computational efficiency—the execution time was the highest, at 208.9 seconds, which may be a limitation in real-time settings.

SVM delivered the strongest overall macro averages (0.94) for precision, recall, and F1-score, confirming its superior ability to distinguish between real and fake news. Precision and recall values were well-balanced for both FAKE and REAL classes, with only minimal variance between them. The high precision for REAL news (0.94) suggests low false positive rates, while the high recall for FAKE news (0.94) indicates strong sensitivity in detecting misinformation. This combination of high accuracy and balanced class performance explains why SVM emerged as the top model in overall evaluation, despite higher computational cost.

3) Naive Bayes

The Naive Bayes classifier exhibited the lowest overall performance, with a training accuracy of 88.63% and a testing accuracy of just 84.77%. It misclassified 193 articles, translating to a misclassification rate of 15.23%, the highest among all models. More concerning was the very high number of false positives (184), suggesting that the model frequently misclassified fake news as real. Although it had a very short execution time of 0.05 seconds, its CV mean score was only 83.43%, with a 2× standard deviation of 1.53%, reinforcing its lower reliability. This model may only be suitable for preliminary baselines or low-resource scenarios.

Naive Bayes presented an imbalanced performance profile. While it achieved an exceptionally high precision for FAKE news (0.98), recall for the same class was notably low (0.71), meaning the model often missed fake news instances. Conversely, REAL news classification showed the opposite trend—very high recall (0.99) but comparatively lower precision (0.77), indicating frequent false positives when labeling articles as REAL. This imbalance is reflected in the lower macro averages (precision: 0.88, recall: 0.85, F1: 0.84). While its near-instant execution time is appealing, this

trade-off in class performance reduces its suitability for high-stakes detection tasks.

4) Random Forest (RF)

Random Forest achieved a perfect training accuracy of 100%, indicating potential overfitting. Its testing accuracy was 91.32%, slightly lower than that of SVM and Logistic Regression. The model misclassified 110 test samples, yielding a misclassification rate of 8.68%, with 70 false positives and 40 false negatives. The CV mean score was 89.15%, accompanied by a 2× standard deviation of 1.06%, indicating reasonably stable performance. The execution time was 9.71 seconds, placing it in the mid-range among tested models. While accurate, its tendency to overfit warrants careful tuning.

Random Forest demonstrated consistent and symmetric precision and recall scores for both FAKE (0.91/0.92) and REAL (0.92/0.91) classes. The macro averages (0.91) suggest stable performance without strong bias toward either class. Its balanced profile means it rarely sacrifices one class's accuracy for the other, making it reliable for varied fake news detection contexts. However, as observed earlier, the perfect training accuracy suggests some overfitting, meaning these results should be interpreted cautiously for unseen data.

5) XGBoost

XGBoost also performed strongly, with a training accuracy of 99.9% and a testing accuracy of 92.03%. It misclassified 101 articles, with a misclassification rate of 7.97%. The model had 67 false positives and 34 false negatives, favoring real news detection slightly. It showed robust generalization with a CV mean score of 91.44% and 2× standard deviation of 1.36%. However, it was relatively slower than Logistic Regression and Random Forest, requiring 56.64 seconds to complete. XGBoost thus provides a good balance between accuracy and generalization but at a higher computational cost.

XGBoost maintained balanced metrics across classes, with macro averages of 0.92 for precision, recall, and F1-score. It achieved slightly higher recall for FAKE news (0.93) than for REAL news (0.91), indicating a mild emphasis on detecting misinformation. The marginally higher precision for REAL news (0.93) compared to FAKE (0.92) also suggests a slight leaning toward reducing false positives for legitimate content. These nuanced strengths make XGBoost appealing for applications prioritizing misinformation detection while minimizing false alarms.

6) Result Summary

Overall, SVM provided the best generalization performance in terms of test accuracy and misclassification rate, while also maintaining balanced precision and recall across both FAKE and REAL classes, though at a high computational cost. Logistic Regression offered a balanced trade-off between accuracy, generalization, efficiency, and consistent performance for both classes. Naive Bayes, while extremely fast, showed an imbalance between precision and recall, reducing its robustness for high-stakes fake news detection. Random Forest and XGBoost performed reasonably well with strong class-wise metrics, but Random Forest showed signs of overfitting and XGBoost was slower than simpler alternatives.

These results highlight that model selection should consider not only accuracy, but also class-wise precision, recall, and F1-score alongside generalization ability and computational efficiency, especially in real-world news classification systems where the impact of false positives and false negatives can differ significantly.

VI. CONCLUSION

This study investigates the capability of several machine learning algorithms in detecting fake news, using textual features derived from the TF-IDF method. The models analyzed represent different categories of learning techniques, including linear models such as Logistic Regression and SVM, a probabilistic approach like Naive Bayes, and ensemble-based methods such as Random Forest and XGBoost. A structured workflow was adopted throughout the study, encompassing exploratory data analysis, text preprocessing, model construction, and performance evaluation. This approach enabled a detailed comparison of the strengths and weaknesses of each classifier in addressing binary classification problems involving real versus fake news.

The experimental results show that SVM and Logistic Regression deliver the most consistent performance, achieving test accuracies of 93.61% and 92.27%, respectively, with relatively low misclassification rates. Notably, SVM demonstrated the highest model stability, as evidenced by the highest average cross-validation score (CV Mean Score) of 93.07%, although at the cost of significantly longer computation time. In contrast, Naive Bayes, while computationally efficient, produced lower accuracy (84.77%) and a higher tendency for false positives (i.e., misclassifying fake news as real), which poses risks in misinformation detection.

Random Forest and XGBoost also performed strongly, with accuracies above 91%. However, they exhibited slightly higher misclassification rates compared to Logistic Regression and SVM. Random Forest achieved 100% accuracy on the training set, indicating potential overfitting, whereas XGBoost offered a balanced trade-off between accuracy, validation stability, and computational efficiency.

Overall, the findings suggest that SVM and Logistic Regression are the most reliable choices for fake news detection tasks based on the dataset used, considering accuracy, misclassification types, and cross-validation stability. The TF-IDF approach effectively captured essential textual features, resulting in a high-dimensional feature matrix (5068 × 39,978) that proved suitable for classification.

Subsequent studies may consider incorporating more sophisticated methods for textual representation, including word embeddings and transformer-based architectures like BERT. Additionally, assessing model effectiveness on multilingual and heterogeneous datasets could enhance the models' ability to generalize across broader contexts.

THANK-YOU NOTE

The authors would like to extend their sincere appreciation to Universitas Amikom Yogyakarta for the ongoing support, research infrastructure, and academic atmosphere that made this study possible. We are especially thankful to the Department of Information Systems, Faculty of Computer Science, for their technical guidance, encouragement, and constructive input throughout the research. We are also grateful to Kaggle for providing access to the Fake News Prediction Dataset, which served as a critical resource for the development and evaluation of our classification models in detecting misinformation.

In addition, we acknowledge the valuable insights and feedback from fellow researchers, academic colleagues, and anonymous reviewers, which greatly enhanced the clarity and rigor of this paper. Lastly, we would like to recognize the global open-source and data science communities whose tools, frameworks, and shared knowledge played a vital role in enabling efficient data preprocessing, modeling, and result visualization, contributing significantly to the reproducibility and reliability of our research outcomes.

REFERENCES

[1] H. Allcott and M. Gentzkow, "Social media and fake news in the 2016 election," *Journal of Economic Perspectives*, vol. 31, no. 2, pp. 211–236, 2017.

[2] S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," *Science*, vol. 359, no. 6380, pp. 1146–1151, 2018.

[3] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake news detection on social media: A data mining perspective," *ACM SIGKDD Explorations Newsletter*, vol. 19, no. 1, pp. 22–36, 2017.

[4] N. Ruchansky, S. Seo, and Y. Liu, "CSI: A hybrid deep model for fake news detection," in *Proc. 2017 ACM Conf. on Information and Knowledge Management (CIKM)*, pp. 797–806, 2017.

[5] K. Shu, S. Wang, and H. Liu, "Beyond news contents: The role of social context for fake news detection," in *Proc. 12th ACM Int. Conf. on Web Search and Data Mining (WSDM)*, pp. 312–320, 2019.

[6] N. J. Conroy, V. L. Rubin, and Y. Chen, "Automatic deception detection: Methods for finding fake news," *Proc. Assoc. for Information Science and Technology*, vol. 52, no. 1, pp. 1–4, 2015.

[7] X. Zhou and R. Zafarani, "Fake news detection: A survey," *ACM Computing Surveys*, vol. 53, no. 5, pp. 1–40, 2019.

[8] H. Ahmed, I. Traore, and S. Saad, "Detecting opinion spams and fake news using text classification," *Security and Privacy*, vol. 1, no. 1, p. e9, 2018.

[9] V. Pérez-Rosas, B. Kleinberg, A. Lefevre, and R. Mihalcea, "Automatic detection of fake news," in *Proc. 27th Int. Conf. on Computational Linguistics (COLING)*, pp. 3391–3401, 2018.

[10] M. Potthast, J. Kiesel, K. Reinartz, J. Bevendorff, and B. Stein, "A stylometric inquiry into hyperpartisan and fake news," in *Proc. 56th Annual Meeting of the Association for Computational Linguistics (ACL), Volume 1: Long Papers*, pp. 231–240, 2018.

[11] R. Binns, "Fairness in machine learning: Lessons from political philosophy," in *Proc. Conf. on Fairness, Accountability and Transparency (FAT)*, pp. 149–159, 2018.

[12] K. Crawford, "Artificial intelligence's white guy problem," *The New York Times*, Jun. 26, 2016. [Online]. Available: https://www.nytimes.com/2016/06/26/opinion/sunday/artificial-intelligences-white-guy-problem.html. [Accessed: Jul. 20, 2025].

[13] J. Ramos, "Using TF-IDF to determine word relevance in document queries," in *Proc. First Instructional Conf. on Machine Learning*, 2003.

[14] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*, 3rd ed. Hoboken, NJ, USA: Wiley, 2013.

[15] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Machine Learning: ECML-98*, C. Nédellec and C. Rouveirol, Eds. Berlin, Heidelberg: Springer, 1998, pp. 137–142.

[16] A. McCallum and K. Nigam, "A comparison of event models for Naive Bayes text classification," in *AAAI-98 Workshop on Learning for Text Categorization*, 1998.

[17] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[18] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, pp. 785–794, 2016.

[19] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proc. 14th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pp. 1137–1143, 1995.