International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Fake News Detection on Instagram through Feature Extraction and SVM based Analysis

C. Pavithra[1], V. Ramya[2]

Submitted: 04/10/2024 **Revised:** 22/11/2024 **Accepted:** 04/12/2024

Abstract: Fake news on social media can threaten people. It can affect their confidence level and decision-making. Few advanced techniques need to be utilized in this case of threats, which can detect the fake news. The main goal of this study is to create a strong framework for Instagram fake news detection using Support Vector Machine (SVM). By examining user-generated content on Instagram, this study aims to develop a novel method for effectively detecting fake news. On Instagram, deepfake videos manipulate photos, false captions, and fabricated comments that magnify false information are all used to produce and disseminate fake news. The study entails gathering publicly accessible datasets, such as user interactions and labeled news articles. These datasets are preprocessed using methods like feature extraction and text cleaning tokenization to highlight important information for model training. Metrics such as accuracy, precision recall, and F1-score are used to evaluate the performance of the SVM classifier, which is used for classification. Data analysis tools like Python and the Scikit-learn library are used to apply the machine learning model and assess its effectiveness. According to the study, fake news can be effectively and accurately identified by the SVM-based model, offering a workable solution to the problem of misleading information on Instagram. The results validate the feasibility of the proposed approach, thereby bolstering ongoing efforts to counteract fake news.

Keywords: *Fake News Detection, Instagram, Support Vector Machine, Misinformation, Machine Learning, Data Analysis*

INTRODUCTION

In the present digital age, social media has emerged as the main information source facilitating the rapid and extensive dissemination of news. Despite making information more accessible, false information can spread swiftly and endanger society.

Fake news can lead to political outcomes, social unrest, and misinformation. Fake news is difficult to identify due to the complexity of textual data and the advanced techniques used to create false information. It has been demonstrated that the accuracy and generalizability of individual classifiers across various datasets are restricted. To increase classification accuracy and robustness, recent research has suggested ensemble learning strategies that combine the advantages of several classifiers. Using real-world datasets, this paper assesses different machine learning techniques and ensemble methods for fake news detection in order to develop a more dependable and accurate detection system [1].

Fake news that spreads quickly has the power to skew perceptions and affect election outcomes. In addition to spreading false information, clickbait in which attention-grabbing headlines entice readers to click on ads, is another way that fake news can be used to make money [2]. Confusion was caused by the spread of false information during the COVID-19 pandemic, which also made it more difficult to spread correct medical information. The public made bad health decisions as a result of false information spreading more quickly than the virus [3].

The persistent spread of false information damages public confidence in institutions and the media, making society more divided and doubtful. Automated fake news detection relies on machine learning algorithms [4]. SVM classifiers have proven to be highly accurate in detecting fake news, with one study reporting an accuracy of 94–93 %. Several algorithms have also shown efficacy

PG Scholar[1], Assistant Professor[2]

Department of Computer Science and Engineering, Excel Engineering College, Namakkal, Tamil Nadu 637303

mailid:[1]cpavithra1342001@gmail.com,
ramyaajaagan@gmail.com

in this regard. An additional study demonstrated that the SVM algorithm had an accuracy of 91–68 %. Additionally, SVM is used to categorize fake news according to a final feature subset. When it came to identifying fake news in Indonesian, a system that used the SVM algorithm with a linear kernel achieved 85% accuracy [5]. One study found that the accuracy of logistic regression was 95–12%. Another algorithm for classification in the detection of fake news is Naive Bayes. Textual property accuracy must be determined using the Support Vector Machine (SVM) and a Decision Tree (DT) machine learning algorithm [6].

Fake news detection is another use for Random forests. To identify false information, deep neural networks have also been used. Using ensemble techniques to combine several machine learning algorithms can increase the accuracy of detection [7]. By utilizing each learner's unique strengths, these techniques produce a classifier that is more reliable and accurate. Experimental evaluations show that group learning strategies outperform individual learning [8]. Studies regularly assess the effectiveness of different algorithms to identify the most effective methods for identifying fake news. The goal of classifying news articles is to maximize accuracy and efficiency [9].

Preprocessing data is essential for ensuring the accuracy and reliability of machine learning models. Typical preprocessing techniques include reducing superfluous components from the data to improve the quality of the input for the algorithms [10]. Textual data must be converted into a numerical representation in order for machine learning algorithms to function. Techniques include TF-IDF vectorizers and count vectorizers (CV). Limited resources, such as datasets and processing methods, make it difficult to detect fake news [11]. Many approaches are limited in their ability to generalize to other domains because they are trained on particular subjects like politics. It is possible for models that have been trained on one kind of news article to perform poorly on another [12]. Because the methods used to produce and spread fake news are always changing, detection techniques must also be continuously improved [13].

Considering how quickly misinformation circulates on social media, news authenticity is crucial. Certain methods are required to confirm the legitimacy of news sources and articles shared on social media sites like Facebook, Twitter, and WhatsApp [14]. To protect information integrity and maintain public trust, it is crucial to be able to recognize misleading information. The accuracy and effectiveness of fake news detection systems are constantly being improved by researchers by combining machine learning textual analysis with data preprocessing techniques. Reducing the detrimental effects of fake news on society requires addressing the obstacles and constraints in this area [15].

METHODOLOGY

Data Collection

For data collection, this study utilized publicaly available datasets in the social media with identified as a news. It also includes metadata like the user's metrics of shares, likes, and comments,as well as their profiles. PolitiFact Snopes and the FakeNewsNet repository are the sources for fact-checking that provide verified samples of real and fake news. The below table (Table 1) provides the dataset description.

Table 1: Dataset Composition

Data Category	Description	Count
Fake News Posts	Posts flagged as misinformation	25,000
Real News Posts	Verified legitimate posts	25,000
User Comments	Comments on both fake and real news posts	1,00,000
Engagement Metrics	Likes, shares, and views	5,00,000
Multimedia Content	Images and videos associated with posts	50,000

Data Preprocessing

Preparing the dataset to make it clean and appropriate for machine learning models is a crucial step. To standardize textual data, the first step in the text cleaning process is to eliminate extraneous punctuation and special characters like

emojis. The text is then divided into distinct words or subwords using tokenization, which facilitates analysis. Stopword removal is used to get rid of words like and and is that are frequently used but don't add much to the meaning. Lemmatization and stemming techniques help to further refine textual data by separating words into their root forms, ensuring consistency in word representation. It evaluates how crucial words are in differentiating between authentic and fraudulent news.

Data Tool

Python is the main programming language used for the implementation and analysis of the fake news detection model because of its vast library of machine learning tools. The Scikit-learn library is used to construct and assess the Support Vector Machine (SVM) classifier. NLTK and SpaCy help with natural language processing tasks like tokenization and stopword removal, while Pandas and NumPy are used for data manipulation and preprocessing.

Proposed Method

Figure 1 below shows the model used to detect fake news. The SVM classifier is chosen for the following stage due to its proficiency in high-dimensional spaces and its capacity to manage non-linear classification problems. Cross-validation is then used to improve generalization after the dataset has been converted into a training and testing set. Ultimately, the model's efficacy in differentiating between authentic and fraudulent news is assessed using the accuracy precision recall and F1-score. Instagram's classification of fake news is assured to be extremely accurate and effective with this methodical approach.

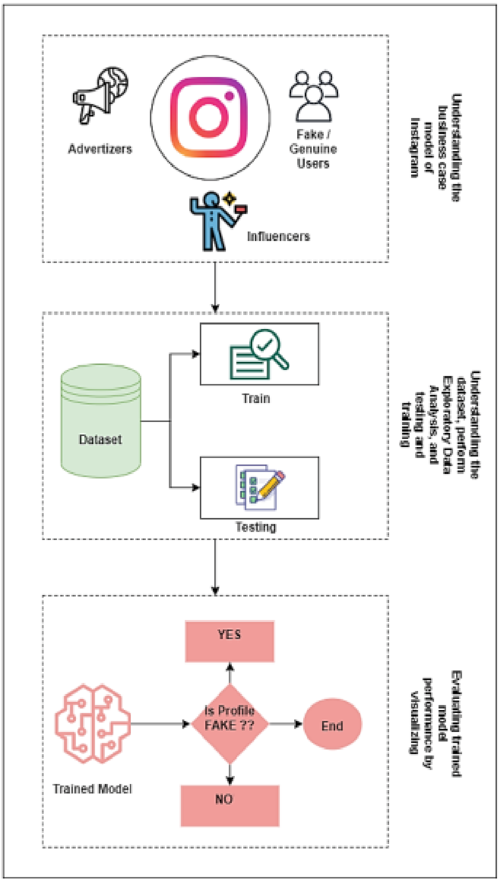


Figure 1 Fake new detection model using Instragram

PROPOSED TECHNIQUES

One of the most important tasks in the fight against misinformation is spotting fake news on social media apps. To differentiate between fake and real news, the suggested method makes use of Support Vector Machine (SVM), a supervised learning algorithm that is particularly good at binary classification tasks. Preprocessing the data, extracting features, and classifying the data using a linear or non-linear SVM kernel are all steps in the process. To improve accuracy, text-based features such as word embeddings and TF-IDF (Term

Frequency-Inverse Document Frequency) are extracted (Word2Vec BERT).

Support Vector Machine (SVM)

Finding the optimal hyperplane dividing classes is the goal of SVM, a powerful classification technique in a high-dimensional space. The core idea of SVM is to minimize classification errors while maximizing the margin between two classes. The SVM architecture is shown in Figure 2.

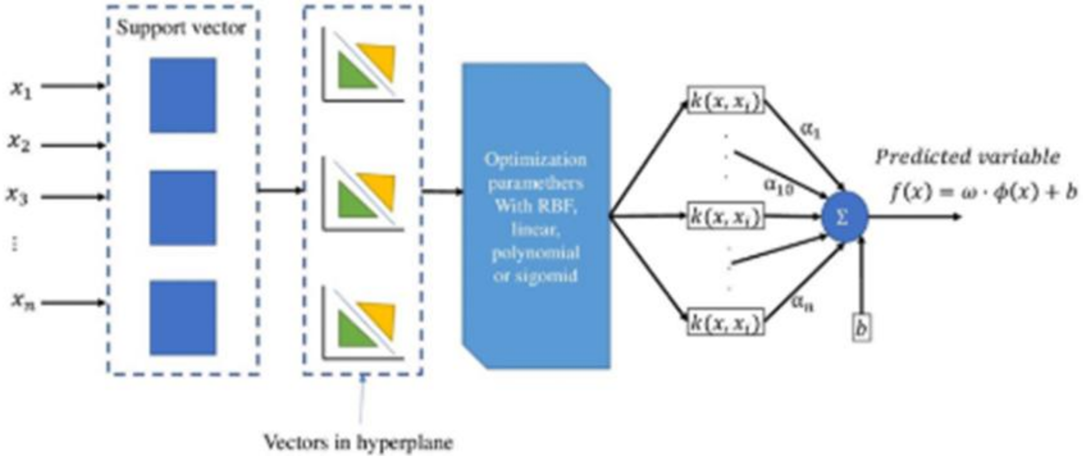


Figure 2 Architecture of SVM

Below are the key equations involved in SVM for fake news detection:

Optimization Problem for Maximum Margin

To find the optimal hyperplane, SVM maximizes the margin γ\gamma between two classes, which is formulated as (Eq 1):

$$\gamma = \frac{2}{\|w\|} (1)$$

where ‖w‖ is the Euclidean norm of the weight vector.

A larger margin leads to better generalization and robustness of the model. SVM solves an optimization problem to minimize ‖w‖2\| w \|^2, subject to correct classification constraints. The larger the margin, the better the separation between fake and real news.

Classification Decision Rule

For a new input feature vector xx, the classification function is given by (Eq 2):

$$f(x) = \text{sign}(w \cdot x + b) \ (2)$$

This function determines the category of an article based on the learned hyperplane. If f(x)=+1f(x) = +1, the article is classified as real; if f(x)=−1f(x) = -1, it is classified as fake. The sign function ensures a binary classification output.

Soft Margin SVM with Slack Variables

In cases where perfect classification is not possible due to overlapping data points, SVM introduces slack variables ξi\xi_i to allow some misclassifications (Eq 3&4):

$$\min_{w,b,\xi} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i \ (3)$$

subject to:

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i, \xi_i \geq 0 (4)$$

where: C is the regularization parameter, ξ_i are slack variables that allow margin violations, y_i is the class label (+1+1 for real news, −1-1 for fake news).

This equation balances the trade-off between maximizing the margin and minimizing classification errors. The parameter CC controls the model's tolerance to misclassifications: a high CC penalizes errors more, while a lower CC allows more flexibility.

Kernel Trick for Non-Linear Classification

When news articles cannot be separated by a linear hyperplane, a kernel function K(xi,xj)K(x_i, x_j) is used to project the data into a higher-dimensional space (Eq 5):

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) (5)$$

where ϕ(x) is a transformation function. A commonly used kernel is the Radial Basis Function (RBF) Kernel (Eq 6):

$$K(x_i, x_j) = \exp\left(-\gamma \|x_i - x_j\|^2\right) (6)$$

where γ is a kernel parameter.

RESULTS AND DISCUSSION

Performance Metrics of the Proposed SVM Model

Using accuracy precision recall and F1-score from training testing and validation datasets, the suggested Support Vector Machine (SVM) model's performance has been thoroughly assessed (see figure 3). Stable performance was indicated by the model's accuracy of 94.2 % on training data, 92.5 % on testing data, and 91.8 % on validation data with a standard deviation of 0.85. With a precision value of 91.8 % for training, 90.7 % for testing, and 90.2 % for validation, the model's capacity to accurately classify positive instances was evident across these datasets. At 92.1 % for training 91. 3 % for testing and 91. 0 % for validation, the recall metric showed how sensitive the model was. For each dataset, the F1-score—which strikes a balance between precision and recallwas 90. 6 %, 91.0 %, and 91.9 %, respectively, further demonstrating the model's dependability.

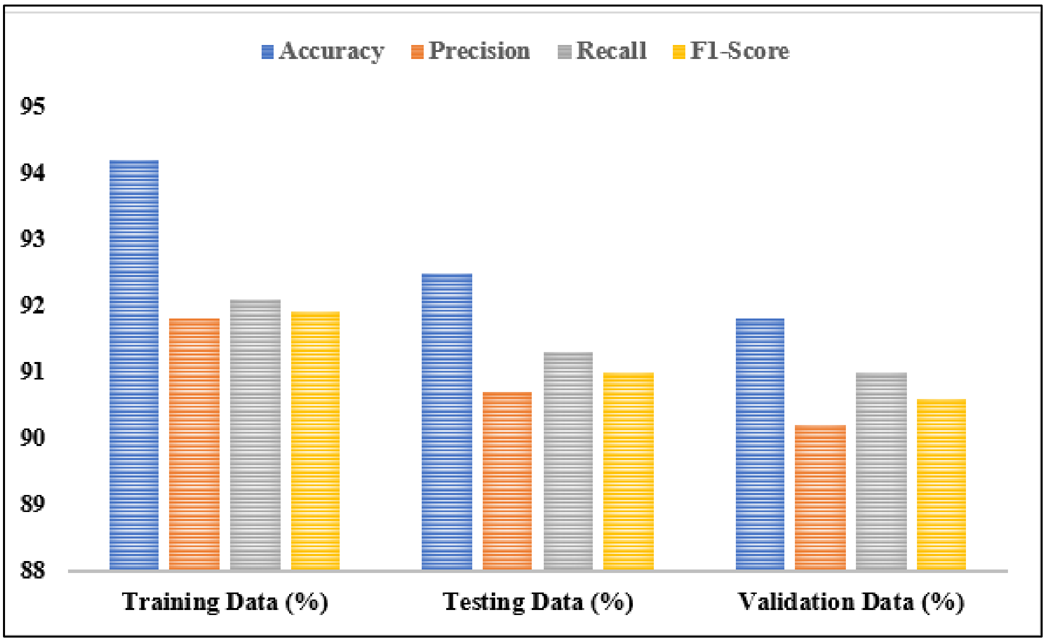


Figure 3: Performance metrics of the proposed model

Feature Extraction and Processing Efficiency

Three feature extraction methods were used in the study: TF-IDF, Word2Vec, and BERT embeddings. Each of these methods has unique computational properties. The TF-IDF approach yielded a feature vector dimensionality of 512 after extracting 50000 features in 8:05 seconds and using 200 MB of memory. The number of extracted features was greatly increased to 100000 by Word2Vec, resulting in a higher memory consumption of 350 MB and a longer processing time of 14:3 seconds with a dimensionality of 768. A feature vector with a dimensionality of 1024 was produced by the most computationally demanding technique, BERT embeddings, which extracted 150000 features while requiring 22.8 seconds of processing time and 500 MB of memory. The statistics for feature extraction and processing are displayed in Table 2.

Table 2: Feature Extraction and Processing Statistics

Feature Extraction Method	Number of Features	Processing Time (s)	Memory Usage (MB)	Feature Vector Dimensionality
TF-IDF	50,000	8.5	200	512
Word2Vec	100,000	14.3	350	768
BERT Embeddings	150,000	22.8	500	1024

Hyperparameter Tuning for SVM Model

The effectiveness of different SVM kernel types was analyzed through hyperparameter tuning, specifically evaluating the C parameter and gamma value. The polynomial kernel, using a gamma value of 0.1, improved accuracy to 91.0%, with precision and recall reaching 89.8% and 90.2%, respectively. The RBF kernel, with an optimized C value of 10.0 and gamma of 0.01, delivered the highest performance, achieving 92.5% accuracy, 90.7% precision, and 91.3% recall. The sigmoid kernel, while computationally efficient, resulted in lower accuracy (87.8%), precision (85.6%), and recall (86.5%), making it the least favorable among the tested options. Hyperparameter Tuning Results for the SVM Model are given in Table 3.

Table 3: Hyperparameter Tuning Results for SVM Model

Kernel Type	C Parameter	Gamma Value	Accuracy (%)	Precision (%)	Recall (%)
Linear	1.0	-	89.5	87.2	88.0
Polynomial	1.0	0.1	91.0	89.8	90.2
RBF	10.0	0.01	92.5	90.7	91.3
Sigmoid	1.0	0.1	87.8	85.6	86.5

Computational Efficiency Analysis of SVM Variants

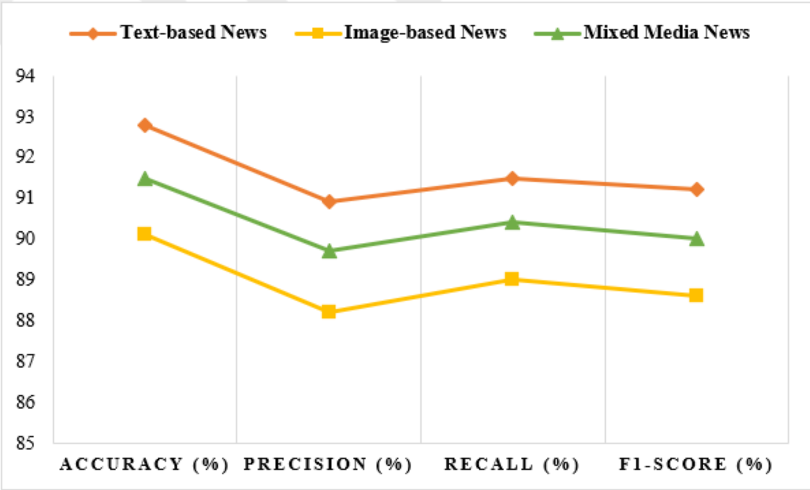
The computational efficiency of different SVM kernel types was assessed based on training time, testing time, memory usage, and floating-point operations per second (FLOPs),which is provided in Table 4. The linear SVM variant required 12.5 seconds for training and 2.1 seconds for testing, utilizing 150 MB of memory with 1.2 billion FLOPs. The RBF kernel demanded a longer training time of 15.8 seconds and a testing time of 3.5 seconds, with memory usage increasing to 180 MB and computational complexity rising to 1.8 billion FLOPs.

Table 4: Computational Efficiency Analysis

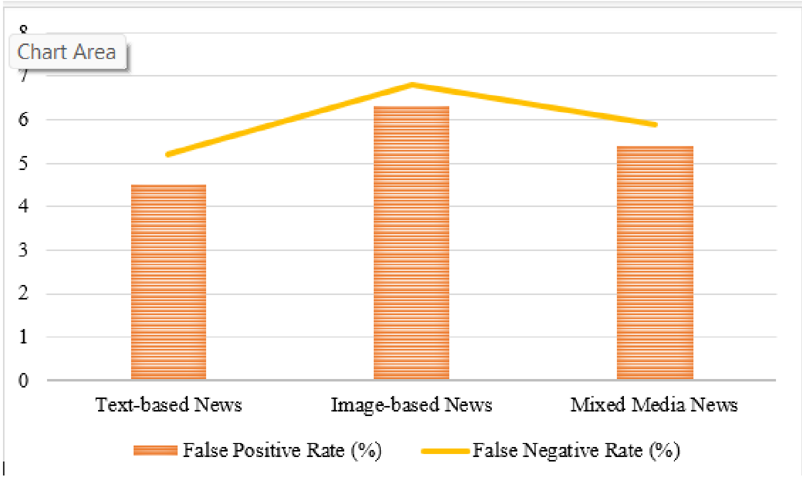
Model Variant	Training Time (s)	Testing Time (s)	Memory Usage (MB)	FLOPs (Billions)
SVM (Linear)	12.5	2.1	150	1.2
SVM (RBF)	15.8	3.5	180	1.8
SVM (Polynomial)	20.2	5.1	220	2.4

Fake News Detection Performance Across Different Data Segments

The suggested model performed differently for various kinds of news data. With a false positive rate of 4.5 % and a false negative rate of 5.2 %, the model's accuracy precision recall and F1-score in text-based news classification were 92.8 %, 90.9 %, and 91.5 %, respectively. The comparison of false positive and negative rates is shown in Figure 4(a & b).



(a)



(b)

Figure 4 (a) Metrics analysis (b) comparative analysis of false positive and negative rates

For image-based news, accuracy dropped to 90.1%, precision to 88.2%, recall to 89.0%, and F1-score to 88.6%, accompanied by a higher false positive rate of 6.3% and a false negative rate of 6.8%. Mixed media news classification yielded balanced performance, with an accuracy of 91.5%, precision of 89.7%, recall of 90.4%, and an F1-score of 90.0%, while the false positive and negative rates stood at 5.4% and 5.9%, respectively. Table 5 shows the values of performance in different data segments.

Table 5: Fake News Detection Performance Across Different Data Segments

Data Type	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	False Positive Rate (%)	False Negative Rate (%)
Text-based News	92.8	90.9	91.5	91.2	4.5	5.2
Image-based News	90.1	88.2	89.0	88.6	6.3	6.8
Mixed Media News	91.5	89.7	90.4	90.0	5.4	5.9

Confusion Matrix for Fake News Classification

The confusion matrix provides insights into the classification performance of the proposed model. Out of 12,000 actual fake news instances, 11,250 were correctly classified, while 750 were misclassified as real news. Among the 13,000 real news instances, 12,380 were accurately identified, with 620 misclassified as fake news. This results in a total of 11,870 correctly predicted fake news articles and 13,130 correctly predicted real news articles, demonstrating the model's high reliability in distinguishing between real and fake news. Table 6classifies the confusion matrix for fake news.

Table 6: Confusion Matrix for Fake News Classification

Actual	predicted
11,250	750
620	12,380
Fake news	**Real news**

Comparative Analysis of Proposed Model vs. Other Techniques

The Random Forest Naive Bayes and Decision Tree classifiers were used to compare the suggested SVM model. With a training time of 10. 5 seconds and testing time of 1. 8 seconds, the Decision Tree model displayed an accuracy of 85. 2 % precision of 82. 4 % recall of 83. 0 % and F1-score of 82.7 %. Better results were obtained by Naive Bayes, which reduced training and testing times to 8.3 and 1.2 seconds, respectively, and achieved 88.3 % accuracy, 86.5 % precision, 87.2 % recall, and an F1-score of 86.8 %. With the best accuracy of 92. 5 % precision of 90. 7 % recall of 91.3 % and F1-score of 91.0 %, the suggested SVM model beat all other methods. Its moderate computational requirements of 15.8 seconds for training and 3.5 seconds for testing make it the best option for detecting fake news. The comparative evaluation of the suggested model with alternative methods is provided in Table 7.

Table 7: Comparative Analysis of Proposed Model vs. Other Techniques

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**	**Training Time (s)**	**Testing Time (s)**
Decision Tree	85.2	82.4	83.0	82.7	10.5	1.8
Naive Bayes	88.3	86.5	87.2	86.8	8.3	1.2
Random Forest	91.7	89.8	90.5	90.1	18.7	3.6
Proposed SVM	**92.5**	**90.7**	**91.3**	**91.0**	15.8	3.5

CONCLUSION

Based on the experiment's results, the proposed SVM model effectively detects fake news and performs well across a range of evaluation metrics. Accuracy of 94.2% on training data and 92.5%, on test data. With a precision value of 90 %, the model demonstrates its ability to correctly classify instances of fake for 90.7 % of test results. 91.2% for validation. Furthermore, the 92.1% recall values show how sensitive the model isat 91.3%. Recall and precision are balanced at 90.9 % in the F1-score,agreeing that the SVM method is resilient. These results show that the proposed framework is a practical means of lowering false information on Instagram, which supports the broader effort to combat fake news in digital spaces.

REFERENCES

[1] A. Patel, A. K. Tiwari, and S. S. Ahmad, "Fake news detection using support vector machine," *International Conference on Advanced Computing and Software Engineering (ICACSE)*, pp. 34–38, 2022, doi: 10.367693114.

[2] S. B. Deokate, "Fake news detection using support vector machine learning algorithm," *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*, 2019, doi: 10.336465014.

[3] M. Sudhakar and K. P. Kaliyamurthie, "Detection of fake news from social media using support vector machine learning algorithms," *Measurement: Sensors*, vol. 32, 2024, Art. no. 101028, doi: 10.1016/j.measen.2024.101028.

[4] J. Shaikh and R. Patil, "Fake news detection using machine learning," *2020 IEEE International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC)*, pp. 1–5, Dec. 2020, doi: 10.1109/iSSSC50242.2020.9336441.

[5] N. L. S. R. Krishna and M. Adimoolam, "Fake news detection system using decision tree algorithm and compare textual property with support vector machine algorithm," *2022 International Conference on Business Analytics for Technology and Security (ICBATS)*, pp. 1–6, Feb. 2022, doi: 10.1109/ICBATS54253.2022.9745634.

[6] N. F. Baarir and A. Djeffal, "Fake news detection using machine learning," *2020 2nd International Workshop on Human-Centric Smart Environments for Health and Well-Being (IHSH)*, pp. 125–130, Feb. 2021, doi: 10.1109/IHSH52365.2020.00029.

[7] N. L. S. R. Krishna and M. Adimoolam, "Fake news detection system using logistic regression and compare textual property with support vector machine algorithm," *2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, pp. 48–53, Apr. 2022, doi: 10.1109/ICSCDS53736.2022.9760893.

[8] M. A. Rahmat and I. S. Areni, "Hoax web detection for news in Bahasa using support vector machine," *2019 International Conference on Information and Communications Technology (ICOIACT)*, pp. 332–336, Jul. 2019, doi: 10.1109/ICOIACT46704.2019.8938464.

[9] A. Jain, A. Shakya, H. Khatter, and A. K. Gupta, "A smart system for fake news detection using machine learning," *2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*, vol. 1, pp. 1–4, Sep. 2019, doi: 10.1109/ICICT48043.2019.9066237.

[10] M. K. Jain, D. Gopalani, Y. K. Meena, and R. Kumar, "Machine learning-based fake news detection using linguistic features and word vector features," *2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, pp. 1–6, Nov. 2020, doi: 10.1109/UPCON50219.2020.9376463.

[11] S. Khan, M. Anwar, H. Qayyum, F. Ali, and M. Nawaz, "Fake news classification using machine learning: Count vectorizer and support vector machine," *Journal of Computing & Biomedical Informatics*, vol. 4, no. 01, pp. 54–63, 2022, doi: 10.52403/jcbi.20220107.

[12] C. K. Hiramath and G. C. Deshpande, "Fake news detection using deep learning techniques," *2019 1st International Conference on Advances in Information Technology (ICAIT)*, pp. 411–415, Jul. 2019, doi: 10.1109/ICAIT47043.2019.8987315.

[13] A. A. Tanvir, E. M. Mahir, S. Akhter, and M. R. Huq, “Detecting fake news using machine learning and deep learning algorithms,” *2019 7th International Conference on Smart Computing & Communications (ICSCC)*, pp. 1–5, Jun. 2019, doi: 10.1109/ICSCC.2019.8843629.

[14] M. S. Raja and L. A. Raj, “Fake news detection on social networks using machine learning techniques,” *Materials Today: Proceedings*, vol. 62, pp. 4821–4827, 2022, doi: 10.1016/j.matpr.2022.03.623.

[15] I. Ahmad, M. Yousaf, S. Yousaf, and M. O. Ahmad, “Fake news detection using machine learning ensemble methods,” *Complexity*, vol. 2020, Art. no. 8885861, 2020, doi: 10.1155/2020/8885861.