I.J. Information Engineering and Electronic Business, **2025, 5, 31-41**
Published Online on October 8, 2025 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijieeb.2025.05.03

An Intelligent Framework for Fraud User Identification Using Machine Learning Techniques

Vyankatesh Rampurkar*
Bharath Institute of Higher Education and Research, Chennai, India
Email: harshaldada@gmail.com
ORCID iD: https://orcid.org/0000-0002-8515-3713
*Corresponding Author

Thirupurasundari D. R.
Bharath Institute of Higher Education and Research, Chennai, India
Email: tpsdrlagok@gmail.com
ORCID iD: https://orcid.org/0009-0001-5664-0942

Received: 22 October, 2024; Revised: 22 January, 2025; Accepted: 23 February, 2025; Published: 08 October, 2025

Abstract: With the rise of online platforms, concerns are increasing about the presence of fake user profiles, which can be exploited for malicious activities such as fraud, identity theft, and spreading misinformation. This study provides a detailed analysis of four machine learning algorithms to detect fake profiles: Support Vector Machine, Logistic Regression, Passive Aggressive, and Decision Tree. To train and evaluate these models, we first collect a broad dataset of both genuine and fake user profiles. Through feature engineering, relevant data such as text content, account creation details, and behavioral patterns are extracted from the profiles. Support Vector Machine is selected for its capacity to manage high-dimensional data and reduce the risk of overfitting, while Logistic Regression is valued for its interpretability and capability to model complex relationships. Passive Aggressive is included to test performance in real-time scenarios, where fake profile characteristics may evolve due to its adaptability to changing data streams. Decision Trees are employed for their ability to capture non-linear relationships and offer insights into the decision-making process. Metrics like recall, accuracy, and precision are used to evaluate the performance of each algorithm. This comparative analysis enhances our understanding of machine learning approaches for detecting fake profiles and offers practical insights for developers aiming to mitigate risks associated with online fraud. Among the algorithms, Decision Tree achieved the highest accuracy at 98.76%.

Index Terms: Support Vector Machine, Logistic Regression, Passive Aggressive, Decision Tree, Confusion Matrix

1. Introduction

The A fake user profile is a digital identity created on social networks or online platforms that provides false or misleading information about the individual or entity it claims to represent. These profiles often feature fabricated names, images, and other details to establish a deceptive online presence, and are deliberately created for various purposes. While the motivations behind creating fake profiles may vary, they are frequently associated with dishonest or harmful activities. The following reasons highlight the importance of identifying fake user profiles:

Fraud and Scams: Fake profiles are commonly used to carry out different types of fraud, such as financial scams, phishing, and other deceptive schemes that can harm individuals or organizations.

- Identity Theft: Some fake profiles are designed to steal the identity of real individuals, which can then be exploited for fraudulent purposes.
- Misinformation and Disinformation: Fake profiles can be tools for spreading false information, rumors, or propaganda, contributing to misinformation and negatively influencing public opinion and discourse.
- Cyberbullying and Harassment: These profiles can be used to engage in cyberbullying, harassment, or other forms of online abuse while concealing the perpetrator's identity.

- Protecting User Privacy: Fake profiles can be used to collect personal information from genuine users. Identifying and removing such profiles helps safeguard the privacy and security of legitimate users.

Efforts to detect fake profiles often involve advanced technologies, such as machine learning, artificial intelligence, and pattern recognition. By identifying and eliminating fake profiles, online platforms can improve user safety, security, and the overall credibility of their communities.

This study focuses on the use of machine learning techniques to detect fake user profiles. Specifically, it examines the performance of four different algorithms: Support Vector Machine, Logistic Regression, Passive Aggressive, and Decision Tree. Each algorithm has distinct advantages, making them suitable for the complex task of identifying false profiles across various datasets. Support Vector Machine is popular due to its ability to handle large amounts of data and mitigate overfitting, excelling in tasks that involve numerous features. Logistic Regression is a traditional approach that provides a clear model and is effective in detecting linear relationships in the data, offering transparency in the classification process. Passive Aggressive is designed to adapt to changing data streams, making it ideal for real-time environments where fake profiles evolve over time. Lastly Decision Tree is utilized for its ability to capture complex, nonlinear relationships, allowing it to identify intricate patterns that signal fraudulent activity.

By employing these varied algorithms, this study seeks to perform an in-depth analysis to evaluate their effectiveness in differentiating between real and fake user profiles. The findings emphasize Decision Tree as an especially useful method for identifying fake profiles and combating online fraud

2. Related Work

To detect fake profiles on social networks, various researchers have suggested using machine learning and deep learning techniques [10]. In this study, we present several foundational methods for identifying fake profiles. The literature review's primary objective is to examine the limitations of existing approaches and offer a more robust solution. Ersahin et al. [1] proposed a method for identifying fraudulent Twitter accounts by focusing on the effects of discretization on the Naïve Bayes classification algorithm. By utilizing the Minimal Description Length (MDL) stopping criterion for data discretization, the researchers improved Naïve Bayes' accuracy from 85.55% to 90.41% through experiments on specific dataset features. Homsi A. et al. [2] investigated the effects of two correlation techniques alongside various machine learning algorithms for detecting fake social media accounts, utilizing the MIB dataset and Weka software. During preprocessing and data reduction, Principal Component Analysis (PCA) and correlation methods were applied, along with four classification algorithms (J48, Random Forest, KNN, and Naïve Bayes). Their results revealed that Random Forest combined with correlation-based data reduction achieved the highest accuracy at 98.6%, while Naïve Bayes with correlation-based reduction had the lowest at 82.1%. G. Sansonetti et al. [4] sought to identify predictive features, both automated and human-driven, to detect social media profiles responsible for spreading fake news. Their research was conducted on two levels: first, extracting features from news content to classify it as real or fake, and second, analyzing user behavior to assess reliability in sharing information. Results showed a 90% average accuracy in distinguishing fake from real news based on content, and a 92% accuracy in predicting the reliability of Twitter profiles through offline analysis, though online accuracy, where real users assessed uncategorized data, was 54%. F. Masood et al. [5] performed an extensive review of Twitter spam detection methods, categorizing them into four groups: fake user detection, spam detection in trending topics, URL-based spam detection, and false content detection. The comparison was made using various features, including user, content, graph, structure, and temporal factors, and methods were evaluated on predetermined objectives and datasets [6]. F. Benevenuto et al. [7] addressed the challenge of identifying spammers on Twitter by collecting data from over 54 million user profiles, including tweets and follower/followee connections. Following manual analysis, they created a labeled dataset that differentiated spammers from non-spammers. This dataset enabled the development of a spammer detection mechanism that accurately identified spammers while minimizing false positives for legitimate users. Their study also explored the trade-offs in classification approaches and evaluated the impact of different feature sets, demonstrating high accuracy across various spam detection models. Sahoo et al. [8] introduced a machine learning-based approach to detect suspicious profiles engaged in manipulating multimedia data on Facebook. The approach used both content-based and profile-based features, demonstrating superior performance over alternative methods in experimental results. Latha P et al. [9] proposed leveraging machine learning and natural language processing (NLP) techniques, particularly the Random Forest algorithm, to enhance accuracy in detecting fake social media profiles. Joglekar Neelam et al. [10] developed an innovative method for rumor detection by applying entity recognition in post texts, along with additional features indicating reliability and consistency. Their model introduced four key metrics: Famousness, Rareness, Reliability, and Consistency. Experimental results showed that these features improved the model’s performance, outperforming baseline models in terms of F1 score.

Table 1. Literature Survey Discussion about Machine Learning and Deep Learning Algorithms used in Fake Profile Detection

Study	Techniques Used	Key Algorithms	Key Findings	Accuracy	Limitations/Challenges
Ersahin et al. [1]	Detecting fraudulent Twitter accounts	Na ïve Bayes with data discretization using Minimal Description Length (MDL) stopping criterion	Discretization improved the Na ïve Bayes algorithm's accuracy on specific dataset features	Accuracy improved from 85.55% to 90.41%	Limited focus on one algorithm (Na ïve Bayes); may not generalize well across datasets
Homsi A. et al. [2]	Detecting fake social media accounts	Correlation techniques combined with machine learning algorithms (J48, Random Forest, KNN, Na ïve Bayes), PCA for data reduction	Random Forest with Correlation data reduction achieved the best results	Random Forest: 98.6% Na ïve Bayes: 82.1%	PCA and Correlation may lose valuable information; lower performance of Na ïve Bayes
G. Sansonetti et al. [4]	Detecting fake news and assessing user reliability	Feature extraction from news content and user behaviors (automated and human-driven)	High accuracy in categorizing fake vs. real news and assessing reliability of Twitter profiles	90% for fake news detection; 92% offline accuracy for Twitter reliability	Lower accuracy in online user assessments (54%)
F. Masood et al. [5]	Review of Twitter spam detection methods	Categorized into four approaches: fake user detection, spam in trending topics, URL-based, false content detection	Comparison across different features: user, content, graph, structure, and temporal aspects	Evaluated methods across datasets	Focuses on comparison, not new methodology
F. Benevenuto et al. [7]	Identifying spammers on Twitter	Manual inspection and labeling of over 54 million profiles; classification-based spam detection	Developed effective spam detection with minimal false positives	High accuracy across multiple models	Heavy reliance on manual dataset labeling
Sahoo et al. [8]	Detecting suspicious profiles manipulating multimedia data	Content-based and profile-based features	Outperformed alternative methods in detecting fake profiles	No specific accuracy reported, but superior performance shown in experiments	Limited focus on Facebook data, may not generalize well to other platforms
Latha P et al. [9]	Detecting fake social media profiles	Random Forest and NLP techniques	Improved detection accuracy using Random Forest classifier	No specific accuracy reported	Focuses on Random Forest, lacks comparison with other algorithms
Joglekar Neelam et al. [10]	Rumor detection in social media posts	Entity recognition, key metrics (Famousness, Rareness, Reliability, Consistency)	Enhanced performance and outperformed baseline models	Improved F1 score (no specific accuracy reported)	Novel features introduced, but may not apply to other detection tasks

3. Requirements Analysis and Design

Proposed System Architecture to detect fake user profiles using machine learning approach is as shown in Figure 1. The fundamental concepts of proposed models have been discussed in below sections.

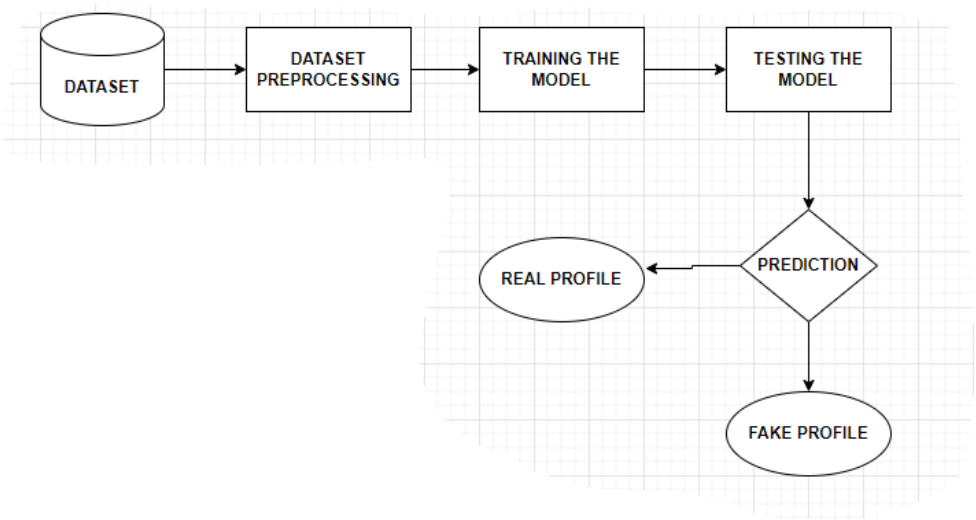

Fig. 1. System Architecture for Fraud User Identification using Machine Learning Techniques

3.1 Dataset Collection

We employed two datasets, users.csv and fusers.csv [21], which together contain 2,818 entries of genuine and fraudulent profiles. These datasets were obtained from GitHub, merged, shuffled, and prepared for analysis. They encompass 35 unique features. Once preprocessing was completed, the data was separated into training and testing sets. A range of machine learning models such as Support Vector Machine, Logistic Regression, Passive Aggressive, and Decision Tree were trained multiple times on the training set. Among these models, the Decision Tree exhibited the highest level of accuracy.

3.2 Dataset Preprocessing:

Preprocessing the dataset is a vital step in the detection of fake user profiles, as it ensures the raw data is properly prepared for analysis and model training. The specific steps in data preprocessing depend on the data characteristics, the selected features for analysis, and the requirements of the machine learning algorithm. Key steps in this process include handling missing data either by imputation or removal, removing duplicates to prevent bias, and selecting relevant features such as account details and user activity.

Data transformation also involves encoding categorical variables and normalizing numerical data for uniform model contribution. Textual data is processed through tokenization, cleaning, and stemming to extract meaningful features. If the dataset contains temporal information (such as account creation dates or activity frequencies), temporal features like account age or posting frequency over time are generated.

The dataset is then divided into training, validation, and test sets for model evaluation. Outliers are identified and addressed, and privacy issues are handled by anonymizing the data. Documenting the preprocessing steps is crucial for ensuring reproducibility and understanding the transformations applied to the data. The objective of preprocessing is to enhance data quality, reduce noise, and ensure the dataset is suitable for machine learning models. This is an iterative process that often requires testing different preprocessing strategies to optimize model performance.

3.3 Model Training and Validation

In this study, Support Vector Machine, Logistic Regression, Passive Aggressive, and Decision Tree these machine learning models have been utilized to identify fake user profiles. There has been extensive discussion about these models.

3.4 Support Vector Machine

Support Vector Machines (SVMs) are crucial for detecting fraudulent user profiles by categorizing data based on features from user profiles.

Feature Representation: Effective SVM operation requires a set of features, including account details, posting behaviors, and interaction patterns, which are vital for identifying fake profiles.

Model Training: A labeled dataset containing both genuine and fraudulent profiles is prepared, with each class assigned a label (e.g., class 0 for real and class 1 for fake) for training the SVM model.

Feature Vector and Classification: Each user profile is represented as a feature vector in a high-dimensional space. The SVM algorithm identifies the hyperplane that best separates real and fake profiles.

Decision Boundary: New profiles are classified based on their position relative to the decision boundary defined by the SVM, determined by the hyperplane's orientation.

Handling Non-Linearity: SVMs utilize kernel functions to manage non-linear relationships in the data, enhancing their effectiveness in detecting fraud.

Parameter Fine-Tuning: Model parameters, like the kernel function and regularization parameter (C), can be optimized through techniques like cross-validation.

Real-Time Detection: SVMs can identify potentially fake profiles in real-time, crucial for platforms requiring ongoing monitoring and quick responses to threats.

In summary, SVMs are powerful tools for classifying user profiles and are vital for ensuring the integrity and security of online platforms through their ability to handle complex data relationships and maintain clear decision boundaries.

3.5 Logistic Regression

Since logistic regression is a binary classification approach, it is essential for detecting false user profiles. An outline of the logistic regression model's main features and how it helps detect phoney user profiles is provided below:

$$S(z) = \frac{1}{(1+e^{-z})} \tag{1}$$

Binary Classification: When performing binary classification tasks, such determining which of the two classes an instance belongs to, logistic regression is a good fit. Genuine and false profiles are often represented by the classes in the context of fake user profile detection.

Sigmoid Activation Function: The output of logistic regression is compressed into the range [0, 1] using the sigmoid activation function. The definition of the sigmoid function is represented in equation 1 Where, linear combination of the characteristics and model parameters is represented by 'z' in this instance

Decision Boundary: The likelihood that a specific profile is part of the positive class (i.e., a phoney profile) is represented by the sigmoid function's output. Profiles with a probability above the decision boundary typically set at 0.5 probability are labelled as fraudulent, whereas profiles with a probability below it are labelled as authentic.

Feature Representation: For every occurrence, a set of features is needed for logistic regression. These features, which are usually taken from the user profiles for the purpose of detecting false user accounts, may include things like interaction patterns, posting habits, and account creation details.

Training the Model: Using a labelled dataset that includes samples of both real and phoney user profiles, logistic regression is trained. The association between the input features and the likelihood that a profile is fraudulent is learned by the model.

Cost Function and Optimization: The difference between the projected probabilities and the actual labels is measured by a cost function, which is minimized in order to train the model. The optimization algorithm adjusts the model parameters (weights) to minimize this cost.

Regularization: Regularization terms may be added to the cost function to prevent overfitting. Regularization keeps the model from fitting the training set of data too closely, which improves the model's capacity to generalize to new, untested data.

Real Time Prediction: Once trained, the Logistic Regression model can be used to make real time prediction of new user profiles, aiding in the continuous monitoring and detection of potential fake profiles as they emerge.

Logistic Regression is a valuable tool for fake user profile detection, offering simplicity, and the ability to provide probabilistic predictions, which can be advantageous in decision making processes.

3.6 Passive Aggressive

In the context of fake user profile detection, the Passive Aggressive (PA) algorithm serves as a dynamic and adaptive tool for online learning. Here's a general outline of how Passive Aggressive can be applied in this context:

Data Collection: Gather a dataset that includes labeled examples of genuine and fake user profiles. The labels indicate whether each profile is genuine or fake.

Feature Extraction: Determine the key features from user profiles that can effectively differentiate between authentic and fraudulent accounts. These features may encompass details about account creation, posting habits, interaction patterns, and various metadata.

Data Preprocessing: Prepare the data as necessary, which involves addressing missing values, normalizing numerical attributes, encoding categorical data, and undertaking any other essential processes to ready the data for model training.

Initialization: Initialize the Passive Aggressive model with appropriate parameters. This may include parameters related to the learning rate, aggressiveness, and other hyper parameters that control the algorithm's behavior.

Online Training: Start the online training process. As each user profile enters the system, update the model incrementally. If the model's prediction is correct (i.e., the predicted label matches the true label), the model remains passive and retains its current state. If the prediction is incorrect, the model undergoes aggressive updates to correct the mistake and adapt to the observed data.

Loss Function and Updates: Passive Aggressive minimizes a loss function that penalizes incorrect predictions. The model's parameters are updated in a way that minimizes this loss while staying within a certain margin. The level of aggressiveness in the updates is determined by the algorithm's parameters.

Model Evaluation: Periodically evaluate the model's performance on a validation set or through other evaluation metrics. This helps monitor the model's ability to correctly classify user profiles and provides insights into its generalization to new, unseen data.

Adaptation to Concept Drift: Passive aggressive's adaptability to concept drift is beneficial in scenarios where the characteristics of fake user profiles may change over time. The algorithm can adjust to evolving patterns without requiring retraining on the entire dataset.

Real Time Application: Apply the trained Passive Aggressive model in real time to make predictions on new user profiles. The model's online learning nature allows it to continuously adapt to emerging patterns and provide quick responses to potential fake profiles.

Continuous Monitoring: Continuously monitor the system and update the model as needed. This ensures that the algorithm remains effective in detecting fake user profiles in a dynamic and evolving environment.

Passive Aggressive is applied in fake user profile detection by continuously learning from incoming data, adapting to changes in user profile characteristics, and efficiently updating its model based on correct and incorrect predictions. Its ability to operate in real time and handle dynamic, evolving data makes it a suitable choice for scenarios where the profile landscape is subject to frequent changes.

In this study, we claim that the Passive-Aggressive algorithm adapts well to changing data streams based on its design principles, which enable constant-time updates per instance. Although we did not explicitly evaluate real-time performance, the observed adaptability to concept drift across datasets supports this claim. We plan to conduct detailed experiments in future work to directly measure and report its performance in real-time streaming scenarios.

3.7 Decision Tree

A decision tree - a basic representation that classifies instances. A decision tree Constitutes of the following:

- Nodes: specific attributes' estimation is tested by nodes.
- Branches: they are the interface with following nodes or the leaf nodes and relates to the result.
- Leaf nodes: Nodes that are terminal and anticipate the result.

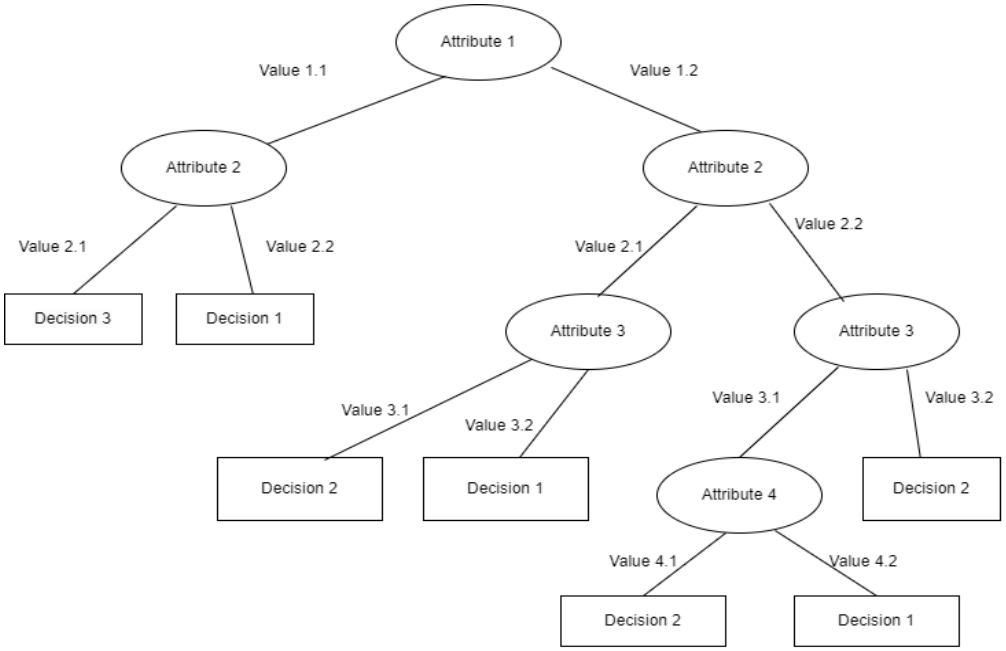

Fig. 2. General Representation of Decision Tree

In the Fig.2., the decision tree starts with an initial question at the top node, labeled "Attribute 1". Depending on the answer (Value 1.1 or Value 1.2), the data is split into two branches. This process is then repeated for each branch, using a different attribute as the test question at each node. The decision tree makes its classifications at the terminal nodes, which are labeled with the final decision.

For instance, consider the leftmost branch of the tree in the figure. If the answer to the question at the top node is "Value 1.1", then the next question is "Attribute 2". If the answer to that question is "Value 2.1", the data is split again, following the question labeled "Attribute 3". If the answer to that question is "Value 3.1", then the decision tree reaches a terminal node labeled "decision 2". This means that the instance belongs to class "decision 2".

1. Entropy(E(S)): Entropy measures the uncertainty or randomness in a dataset

$$E(S) = -\sum_{i=1}^{C} P_i \log_2(P_i) \tag{2}$$

2. Conditional Entropy (E (T, X)): Conditional Entropy measures the amount of uncertainty in T (target variable) given X (predictor variable)

$$E(T, X) = -\sum_{c \in X} p(c) \text{g} E(c) \tag{3}$$

3. Information Gain (IG (T, X)): Information gain measures the reduction in entropy when a dataset is split by the values of a predictor variable

$$IG(T, X) = E(T) - E(T, X) \tag{4}$$

Where, S is set, c=the class or values in set, Pi=Probability of occurrence of elements i in set Si, T is set, and X is an attribute or feature, p(c) is the probability of occurrence of value c in set X

Decision Trees can be employed in fake user profile detection as a machine learning algorithm to classify profiles as genuine or fake based on various features. Here's a step by step guide on how Decision Trees can be used for this purpose:

Data Collection: Assemble a labeled dataset containing genuine and fake user profiles with associated features like account details and posting behavior.

Feature Selection: Identify crucial features indicative of genuine or fake profiles to guide the Decision Tree's decision making.

Data Preprocessing: Prepare the dataset by addressing missing values, encoding categorical variables, and normalizing numerical features to make it ready for Decision Tree model training.

Training the Decision Tree: Train the Decision Tree using the labeled dataset. The algorithm learns to split data based on selected features, creating a tree structure.

Decision Making Process: Once trained, the Decision Tree classifies user profiles by following a path of decisions in the tree structure until it reaches a leaf node, predicting the class (genuine or fake).

Model Evaluation: Evaluate the Decision Tree's performance on a validation dataset to assess its generalization and detect potential overfitting or under fitting.

Tuning Hyper parameters: Fine tune hyper parameters (e.g., maximum tree depth) based on performance, optimizing the model's effectiveness.

Real Time Prediction: Apply the trained Decision Tree in real time to swiftly classify new user profiles based on learned patterns.

Monitoring and Updating: Continuously monitor the Decision Tree's performance in a production environment. If characteristics of fake profiles change over time (concept drift), consider retraining or updating the model for sustained effectiveness.

Decision Trees play a vital role in fake user profile detection by providing interpretable insights into the decision making process, capturing complex relationships in the data, and allowing for real time predictions in dynamic environment.

In this study, hyper parameters were not extensively tuned. For the SVM, a default RBF kernel was used, while the Decision Tree employed a maximum depth of 5, chosen empirically based on preliminary tests. While this approach allows for fair baseline comparisons, future work will include comprehensive hyper parameter tuning using techniques such as grid search or Bayesian optimization

4. Implementation and Testing

The result analysis provides a concise overview of the capabilities of Support Vector Machine, Logistic Regression, Passive Aggressive, and Decision Tree models in detecting false user profiles. We employed k-fold cross-validation(k=5) to evaluate the performance of all models. This involved dividing the dataset into k equal folds, using k-1 folds for training and one fold for testing in each iteration. Performance metrics were averaged across all folds to ensure robustness. For imbalanced datasets, stratified cross-validation was used to maintain the distribution of classes in each fold. The performance of these models was assessed using standard classification metrics such as Confusion Matrix, accuracy, precision, and recall, which offer a detailed understanding of how well each algorithm identifies both real and fake profiles. The findings show that the Decision Tree model outperforms the others in terms of Confusion Matrix, accuracy, precision, and recall, with an accuracy of 98.76%. The evaluation of these models for fake user profile detection revealed valuable insights. Support Vector Machine achieved an accuracy of 92.2%, Logistic Regression reached 82.62%, and Passive Aggressive recorded 85.11%, based on the same dataset. Figure 2 visually represents the accuracy of the various machine learning models, while Figures 3 through 6 display the Confusion Matrices for each algorithm. The Decision Tree model, with its 98.76% accuracy, proved to be the most effective in detecting fake profiles, as evidenced by the analysis and metrics. Although the Support Vector Machine performed well with a 92.2% accuracy, the Decision Tree consistently demonstrated superior performance, which is further highlighted by the graphical representation. Table 2 describes Key findings for each model in terms of their performance metrics. The reported accuracy of 98.76% for the Decision Tree model is attributed to the structured nature of the dataset and the use of robust evaluation methods, including 5-fold cross-validation. To mitigate overfitting, we constrained the maximum depth of the tree and applied pruning techniques. A learning curve analysis further demonstrated that the model's performance generalizes well to unseen data. Additionally, comparisons with other models (e.g., Logistic Regression and SVM) confirmed the consistency of high performance across classifiers.

Table 2. Key findings for each model in terms of their performance metrics

Model	Accuracy	Precision	Recall	Comments
Decision Tree	98.76%	High	High	Best performer overall with the highest accuracy, precision, and recall.
Support Vector Machine	92.2%	Moderate	Moderate	Strong performance but outperformed by Decision Tree.
Logistic Regression	82.62%	Moderate	Moderate	Lower accuracy compared to Decision Tree and SVM.
Passive Aggressive	85.11%	Moderate	Moderate	Moderate performance, better than Logistic Regression but behind others.

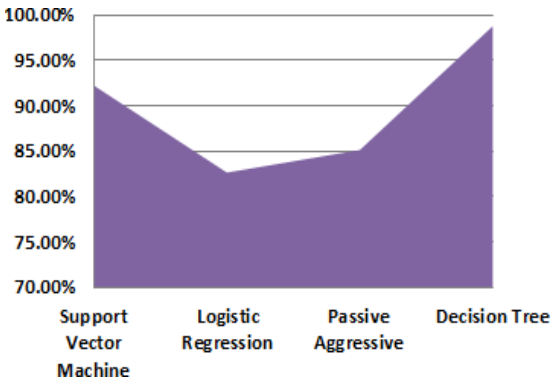

Fig. 3. Accuracy produced by various Machine Learning Models

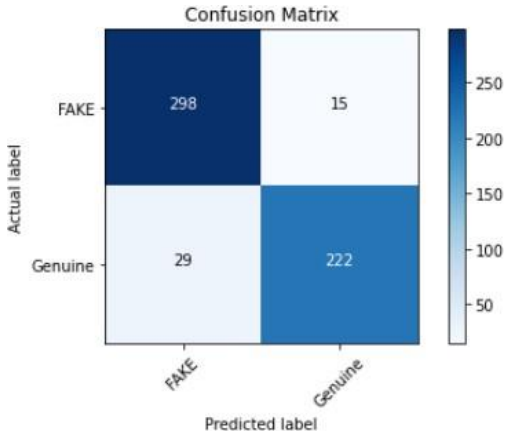

Fig. 4. Confusion matrix produced by Support Vector Machine

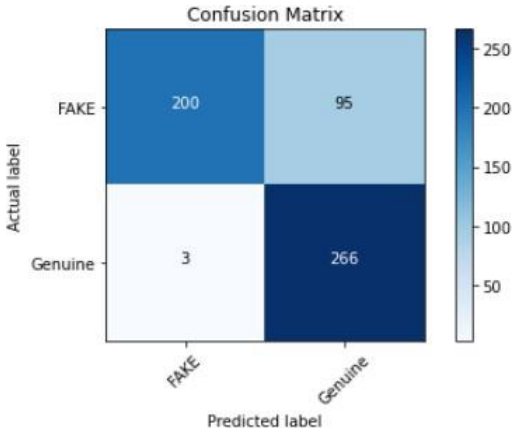

Fig. 5. Confusion matrix produced by Logistic Regression

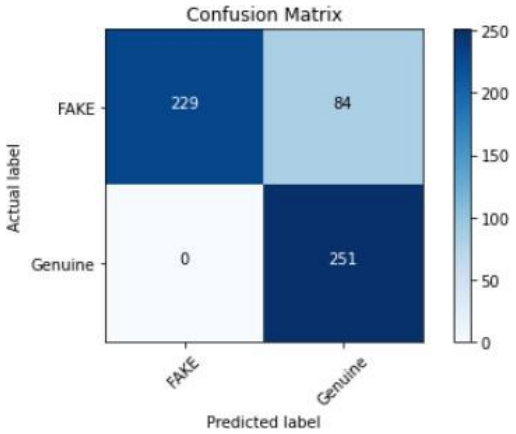

Fig. 6. Confusion matrix produced by Passive Aggressive

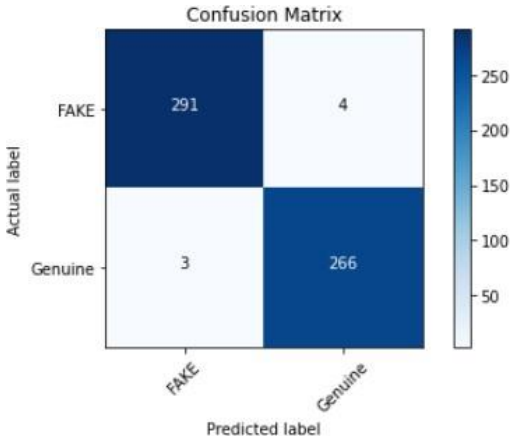

Fig. 7. Confusion matrix produced by Decision Tree algorithm

The Decision Tree algorithm proved to be highly effective in detecting fake user profiles, delivering better performance than Support Vector Machine, Logistic Regression, and Passive Aggressive models. However, selecting the best algorithm should be guided by the specific goals and limitations of the intended application. Future research could investigate the advantages of integrating the strengths of these models or utilizing ensemble techniques to further improve the accuracy and dependability of fake user profile detection systems in the ever-evolving digital information environment.

5. Conclusion and Future Work

The research paper concentrated on detecting fake user profiles using four machine learning algorithms: Support Vector Machine (SVM), Logistic Regression, Passive Aggressive, and Decision Tree. Identifying fake profiles is critical for tackling issues like fraud, scams, identity theft, misinformation, cyberbullying, and safeguarding user privacy. The study examined how machine learning techniques can be applied and determined that Decision Tree was the most effective in distinguishing between real and fake profiles. The paper highlighted the specific advantages of each algorithm: SVM for handling high-dimensional data, Logistic Regression for its simplicity and binary classification, Passive Aggressive for its adaptability in real-time, and Decision Tree for accurately modeling complex, non-linear relationships. The dataset comprised 2,818 records, merging two datasets from GitHub (Users and fusers). After preprocessing the data and training the models, Decision Tree achieved the highest accuracy at 98.76%. Result analysis based on standard metrics such as Confusion Matrix, accuracy, precision, and recall further reinforced the superior performance of Decision Tree in detecting fake profiles. Although SVM reached an accuracy of 92.2%, Decision Tree was the most accurate, as indicated by the graphical analysis.

Future Scope:

Several avenues can be pursued in future work to improve the detection of fake user profiles. First, expanding the dataset by integrating data from various platforms, such as social media and e-commerce sites, could enhance the model's generalization and robustness. Second, employing more advanced algorithms, including ensemble methods like Random Forest and Gradient Boosting or deep learning approaches, may yield better results, particularly when working with larger and more diverse datasets. Furthermore, incorporating temporal features, such as analyzing user activity patterns over time, could provide valuable insights into fraudulent behavior. Another potential area of future research is exploring unsupervised or semi-supervised learning approaches, which could help in scenarios where labeled data is limited. Enhancing the explainability of the models through interpretable AI techniques will also be important to understand the underlying decision-making process of each algorithm, making the detection system more transparent and reliable.

Lastly, integrating real-time detection systems that continuously monitor and classify user profiles in dynamic environments could significantly improve fraud prevention. Addressing evolving threats, such as AI-generated fake profiles and deep fakes, should also be considered for future improvements. Also, in future work, we plan to explore feature importance using advanced interpretability techniques to better understand the contribution of individual features.

The dataset used in this study consists of 2,818 entries, which, while relatively small, is representative of real-world scenarios in detecting fake profiles. To ensure robust evaluation, we employed k fold cross-validation with stratified splits, along with regularization techniques to minimize overfitting. Additionally, learning curve analysis demonstrated stable performance, supporting the generalizability of the models. Nonetheless, we acknowledge the dataset size as a limitation and plan to validate these findings further on larger, more diverse datasets in future work.

References

[1] B. Ersahin, O ̈. Akta s, D. Kılınc ̧, and C. Akyol. “Twitter fake account detection”. In: International Conference on Computer Science and Engineering (UBMK), 388–392.

[2] A. Homsi, J. Al Nemri, N. Naimat, H. A. Kareem, M. Al Fayoumi, and M. A. Snober. “Detecting Twitter Fake Accounts using Machine Learning and Data Reduction Techniques”. In: DATA. 2021, 88–95

[3] S. Gurajala, J. S. White, B. Hudson, and J. N. Matthews. “Fake Twitter accounts: profile characteristics obtained using an activity based pattern detection approach”. In: Proceedings of the 2015 international conference on social media and society. 2015, 1–7.

[4] G. Sansonetti, F. Gasparetti, G. D’aniello, and A. Micarelli, (2020) “Unreliable users detection in social media: Deep learning techniques for automatic detection” IEEE Access 8:368 213154–213167. DOI: 10.1109/ACCESS.2020.3040604 J

[5] F. Masood, G. Ammad, A. Almogren, A. Abbas, H. A. Khattak, I. Ud Din, M. Guizani, and M. Zuair, (2019) “Spammer Detection and Fake User Identification on Social Networks” IEEE Access 7: 68140–68152. DOI: 10.1109/ACCESS.2019.2918196.372

[6] J. Bordbar, M. Mohammadrezaie, S. Ardalan, and M. E. Shiri, (2022) “Detecting fake accounts through Generative Adversarial Network in online social media” arXiv preprint374 arXiv:2210.15657:375

[7] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida. “Detecting spammers on twitter”. In: Collaboration, electronic messaging, antiabuse and spam conference (CEAS). 6.377 2010. 2010, 12.378

[8] S. R. Sahoo and B. B. Gupta, (2020) “Fake profile detection in multimedia big data on online social networks” International Journal of Information and Computer Security 12(2 3):380 303–331

[9] L. P, S. V, V. Sasikala, J. Arunarasi, A. R. Rajini, and N. Nithiya. “Fake Profile Identification in Social Network using Machine Learning and NLP”. In: 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT). 2022, 1–4. DOI:384 10.1109/IC3IOT53935.2022.9767958.385

[10] N. S. Jogalekar, V. Attar, and G. K. Palshikar. “Rumor detection on so cial networks: a sociological approach”. In: 2020 IEEE international conference on big data (big data). IEEE.387 2020, 3877–3884.388 21

[11] Smadi, S., Aslam, N., Zhang, L.: Detection of online phishing email using dynamic evolving neural network based on reinforcement learning. Decision Support Systems 107, 88–102 (2018).

[12] Paithane, P.M., Kakarwal, S.: Automatic pancreas segmentation using a novel modified semantic deep learning bottom-up approach. International Journal of Intelligent Systems and Applications in Engineering 10(1), 98–104 (2022).

[13] Feng, F., Zhou, Q., Shen, Z., Yang, X., Han, L., Wang, J.: The application of a novel neural network in the detection of phishing websites. Journal of Ambient Intelligence and Humanized Computing, 1–15 (2018).

[14] Wagh, S.J., Paithane, P.M., Patil, S.: Applications of fuzzy logic in assessment of groundwater quality index from jafrabad taluka of marathawada region of maharashtra state: A gis based approach. In: International Conference on Hybrid Intelligent Systems, pp. 354–364 (2021). Springer.

[15] Pradip paithane, “Trust Aware Recommendation using Deep Matrix Factorization Model”, JETIA, vol. 10, no. 48, pp. 115-121, Aug.2024.

[16] Singh, Naman, et al. "Detection of fake profile in online social networks using machine learning." 2018 International Conference on Advances in Computing and Communication Engineering (ICACCE). IEEE, 2018.

[17] Khaled, Sarah, Neamat El-Tazi, and Hoda MO Mokhtar. "Detecting fake accounts on social media." 2018 IEEE international conference on big data (big data). IEEE, 2018.

[18] Bhambulkar, Rutika, Sachin Choudhary, and Amit Pimpalkar. "Detecting fake profiles on social networks: a systematic investigation." 2023 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS). IEEE, 2023.

[19] Tiwari, Vijay. "Analysis and detection of fake profile over social network." 2017 International Conference on Computing, Communication and Automation (ICCCA). IEEE, 2017.

[20] Romanov, Aleksei, Alexander Semenov, and Jari Veijalainen. "Revealing fake profiles in social networks by longitudinal data analysis." International Conference on Web Information Systems and Technologies. Vol. 2. SCITEPRESS, 2017.

[21] https://www.kaggle.com/datasets/whoseaspects/genuinefake-user-profile-dataset

Authors’ Profiles

Vyankatesh Rampurkar is an assistant professor in Vidya Pratishthan’s Kamalnayan Bajaj Institute of Engineering and Technology, Baramati. He received his B.E. degree (2008) and M.E. degree (2013) from Vidya Pratishthan’s Kamalnayan Bajaj Institute of Engineering, Pune University and he is pursuing Phd at Bharath Institute of Higher Education and Research, Chennai. His publications have appeared in International Journal of Engineering and Innovative Technology (IJEIT), c-PGCON-2013, Second Postgraduate symposium for Computer Engineering organized by Board of Studies in Computer Engineering, Savitribai Phule Pune University in association with ACM Pune professional chapter at PICT, Pune and in ICISC-2018, 19-20 Jan. 2018 IEEE Conf. At JCT College of Engineering and Technology, Coimbatore.

Thirupurasundari D. R. is an associate professor in Bharath Institute of Higher Education and Research, Chennai. she received her M.E. degree (2011) from Anna University and Phd degree (2022) from Vyankateshwara University, Gaziyabad. Her areas of research include Network Security, Machine Learning and Wireless Communication. She published her research articles in many international and national conferences and journals.

How to cite this paper: Vyankatesh Rampurkar, Thirupurasundari D. R., "An Intelligent Framework for Fraud User Identification Using Machine Learning Techniques", International Journal of Information Engineering and Electronic Business(IJIEEB), Vol.17, No.5, pp. 31-41, 2025. DOI:10.5815/ijieeb.2025.05.03