Fake Instagram Account Detection Using Single and Stacking Ensemble Models on Integrated Static and Temporal Features

Samar Sadiq Hassoun[1]* and Nahla A. Flayh[2]

[1,2]Department of Information Computer Systems, College of Computer Science and Information Technology, Basra University, IRAQ.

*Corresponding Author: Samar Sadeq Hassoun

ABSTRACT: This research examines the detection of fake Instagram accounts by comparing single machine learning classifiers with a stacking ensemble model applied to integrated static and temporal features. A dataset of 7,594 accounts was collected over five months using multiple scraping tools, ensuring a balanced representation of real and fake profiles. Key behavioral, engagement, and account-level features were extracted to capture users' activity patterns. Among the individual classifiers, Random Forest achieved the highest performance; however, the stacking ensemble model outperformed all single models, reaching an accuracy of 0.9888 with superior precision, recall, and F-measure values. The findings demonstrate that integrating temporal metrics with ensemble learning significantly enhances detection accuracy and provides a more reliable approach for identifying fraudulent accounts on social media platforms.

Keywords: Machine Learning, Fake Account Detection, Stacking Ensemble; Temporal Features.

1. INTRODUCTION

Instagram is a widespread source of content and marketing with numerous easy methods of account creation that enable fake accounts to manipulate the engagement or promulgation of misinformation, presenting difficulties in the dirty side of cybersecurity and social media intelligence [1]. Most previous studies have been evaluating an easy aspect, such as a follower count and bio, and not paying close attention to the dynamic behavioral patterns. This paper explores the factors that are constant and dynamic, including the frequency of posts, variation in activities, and unusual follower numbers, to enhance the detection of fake accounts. Since the appropriate datasets to analyze the convergence of two metrics over time are not available, we have developed our dataset by using automated and manual collection and chronological analysis. We also use a stacking ensemble model, a model that integrates a variety of classifiers with the help of a meta-learner to increase the accuracy, decrease overfitting, and improve the detection process [2].

2. Related Work

Several studies have addressed Instagram fake account detection using different datasets, feature sets, and classifiers: In [3], "Detection of Fake Accounts in Instagram using Machine Learning" applied Logistic Regression (LR) and Random Forest (RF) on a Kaggle dataset of 696 profiles with 11 static user activity and profile features. The Random Forest achieved an accuracy of 92.5%. In [4], "Instagram Fake and Automated Account Detection" employed Naive Bayes, LR, SVM, and Neural Networks to distinguish genuine and fake accounts. Two datasets were used: one of 1,203 manually labeled profiles with eight static features (accuracy 94%) and another of 1,400 profiles with nine statistical and eight temporal features collected over six months (accuracy 86%). In [5], "Classification of Instagram Fake Users Using Supervised Machine Learning Algorithms" applied RF, Multi-Layer Perceptron (MLP), LR, Naive Bayes, and J48 Decision Tree on 65,326 profiles. Real users were gathered from followers of 24 Indonesian private university pages, while fake users were bought from sellers. Using 16 static and one temporal feature, the highest accuracy was 90.09%. In [6],

"Detecting Fake Instagram Profiles: A Novel Machine Learning-Based Framework" used a deep neural network with Insta Fake and Insta Reach tools on 576 Kaggle profiles with 11 static attributes, achieving 91% accuracy. In addition, in [7], "Machine Learning Interpretability to Find Fake Accounts in Instagram" tested RF, XG Boost, AdaBoost, SVM, Decision Tree (DT), and KNN on 696 Kaggle profiles using 11 static features. The best performance reached 96% accuracy. In [8], "Countering Social Media Cybercrime Using Deep Learning: Instagram Fake Accounts Detection" employed LSTM on two datasets: 1,203 GitHub profiles with eight static features (accuracy 97.42%) and 696 Kaggle profiles with 11 static features (accuracy 94.21%). The other contribution is done by [9]. "Detection of Fake Instagram Accounts via Machine Learning Techniques" applied RF, LR, MLP, KNN, DT, SVM, and Gaussian Naive Bayes on two datasets: 1,194 GitHub profiles with eight static features and 696 Kaggle profiles with 11 features. Combined into 1,890 profiles with three additional features, the scheme achieved 97.36% accuracy. Also, in [10], "An Efficient Method for the Instagram Platform" evaluated multiple classifiers, including DT, LR, SVM, RF, KNN, XG Boost, Gradient Boosting, AdaBoost, and Extra Trees on 65,326 Kaggle profiles with 16 static and one temporal feature, reaching 90% accuracy.

3. METHOD

The proposed Instagram fake account detection model employs a structured data collection methodology and utilizes multiple base classifiers combined with a meta-learner. The workflow of the model consists of several key stages, beginning with data collection, followed by feature extraction and preprocessing, and concluding with model training and performance evaluation. An overview of this process is presented in Fig. 1.

3.1 Data Collection and Feature Extraction

The data collected in the study was compiled through a hybrid multi-source technique so as to guarantee diversity, reliability, and completeness. There were 7,594 Instagram accounts, and 3,832 and 3,762 fake and real accounts, respectively. Data were collected complementarily through the use of four tools, specifically the Instagram Graph API, which provided structured and verified account information and contributed 2,995 accounts; web scraping tools, such as Scraper API, Insta Loader, Selenium, and Beautiful Soup, which contributed 3,523 accounts; the Google Custom Search API and Apify platform, which provided 1,000 accounts; and manual data collection, which served to check the validity of the results and inclusion of various samples (contributing 76 accounts). The data has 11 fixed characteristics, 5 time characteristics, and a label that states that the account is real or fake. The dataset enables a comparative evaluation between models trained on static features alone and those trained on a hybrid set including both static and temporal features, aiming to assess whether the incorporation of temporal behavior improves the detection accuracy of fake Instagram accounts. Table 1 summarizes the complete dataset, and Fig. 2 illustrates the percentage distribution of real and fake accounts.
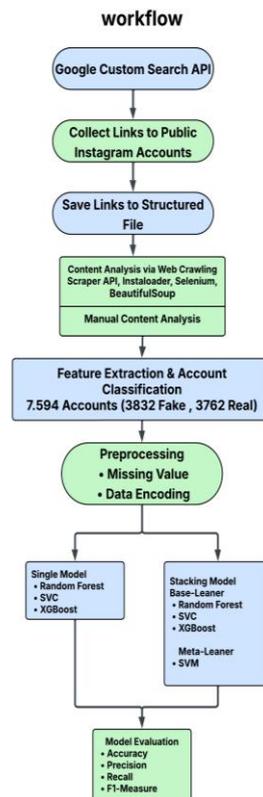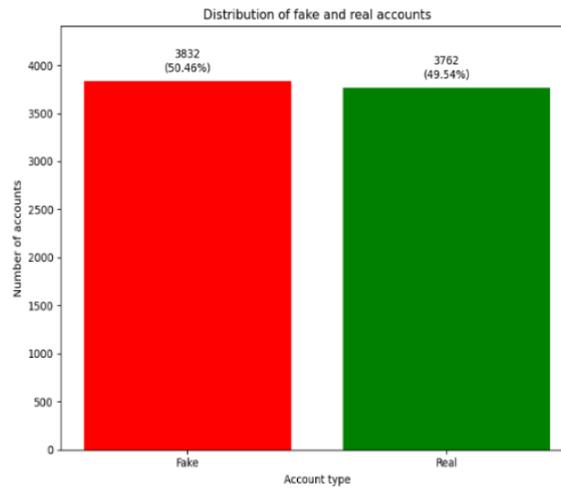

FIGURE 1. - Data Collection and Feature Extraction Workflow

Figure 2. - Distribution of Fake and Real Accounts in The Dataset

Table 1. - Summary of The Dataset (Static and Temporal Features)

NO	Features	Description	Feature Type
1	Profile Pic	Determines whether an account has a profile picture or not. Fake accounts often lack a clear picture.	Static
2	Numbers/Length Username Ratio	Measures the frequency of numbers in the username. Fake accounts often use many random numbers.	Static
3	Number of Words in Full Name	Indicates the number of words in the account's full name.	Static
4	Numbers/Length Full Name Ratio	Measures the ratio of numbers to the length of the full name. Fake accounts tend to have a higher ratio of numbers.	Static
5	Full Name/Username Match	Checks whether the full name matches the username. This is common in fake accounts.	Static
6	Description Length	Measures the number of characters in the account's description. Fake accounts typically have shorter descriptions.	Static
7	External URL	Indicates the presence of an external link in the profile. Fake accounts may use links for promotional purposes.	Static
8	Account Status (Private)	Indicates whether the account is private or public. Some fake accounts start as public and later become private.	Static
9	Number of Posts	Reflects the account's activity through the number of posts shared.	Static
10	Follower Count	Shows the number of followers an account has. An unusually high number may indicate abnormal account activity.	Static
11	Number of Accounts Followed	Indicates how many accounts a user follows. Fake accounts may follow a strange number of accounts or engage in bot-like behavior.	Static
12	Average Comments per Post	Represents the average number of comments received on each post, reflecting engagement.	Temporal
13	Average Likes Per Post	Indicates the average number of likes per post	Temporal

14	Followers/Following Ratio	Measures the proportion of followers to following; higher values may indicate influence	Temporal
15	Daily Follower Change	Captures the average daily variation in follower count, useful for detecting anomalies	Temporal
16	Posting Consistency	Assesses the regularity of posting activity over time, important for behavioral analysis	Temporal
17	Flag Label	Used to mark suspicious or abnormal accounts in the dataset	

3.2 Feature Analysis

This section presents an analytical examination of the features utilized in the study, emphasizing their statistical characteristics, relevance, and interrelationships. Conducting this analysis is crucial, as it enhances both the reliability and the interpretability of the machine learning models.

3.2.1 Correlation Analysis

Determining the relationship between variables is based on the most common attributes in a given task. In achieving this goal a correlation matrix was employed to try to find the most correlated attributes for a given dataset. These attributes are sometimes referred to as superfluous [6]. This analysis was conducted and the outcome was represented in form of the Heatmap diagram presented below in Fig. 3. The figure is indicating how strong the relationships of the pairs of features are by the colors.
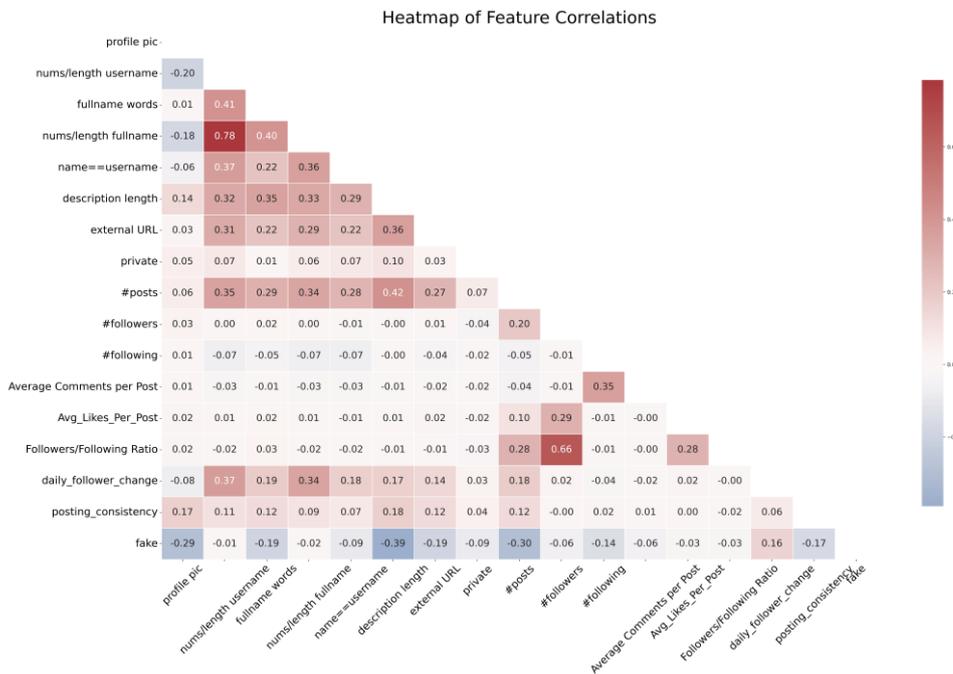

FIGURE 3. - The heat map diagram

3.2.2 Feature Importance Evaluation Using Random Forest

A random forest approach was utilized to evaluate the features' importance toward Instagram account classification. It estimated the classification process's features and allowed us to ranked them in order of importance. This analysis shed light on what attributes influenced an account's classification as real or fake the most. Fig. 4 demonstrates the features' relative importance.
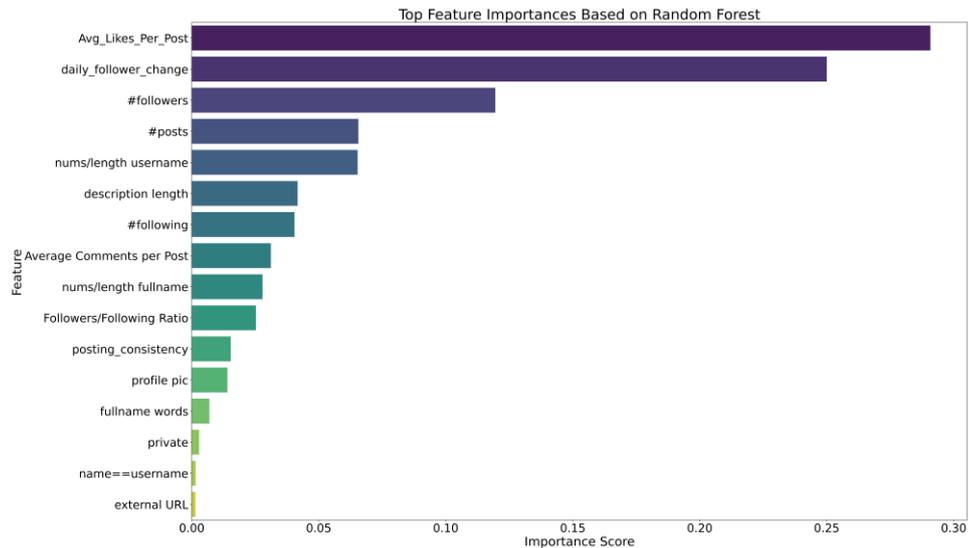
FIGURE 4. - The Feature Importance and Redundancy

3.3 Preprocessing

For this research, the Instagram dataset was preprocessed in two main steps:

- Handling missing data: The median was used to fill in gaps for missing values in the numerical columns [11] so as to maintain the integrity of the data set.

- Categorical Data Encrypting: Categorical data was transformed into numbers through the label encoding method to fit machine learning models [12].

Preparation steps have not changed the distribution of the category of the data group. Both fake and real account categories have maintained their original balance, ensuring the maintenance of the data group and that the training process was not affected by the class imbalance due to the quality of the data.

3.4 Experimental Setup and Data Splitting

In an attempt to assess the performance of the proposed machine learning models, the dataset of 7,594 Instagram accounts was split into two parts based on a hold-out validation strategy. In particular, 80 percent of the data (6,075 accounts) was devoted to the model training, and the other 20 percent (1,519 accounts) was distributed to the testing. In order to be able to reproduce the experimental results, the process of data splitting was performed with a constant random seed (random_state = 42). Since there was an almost equal representation of real and fake accounts in the dataset, class balances are maintained in both training and testing subsets, which allows making the necessary assessment of the classification models without any bias.

3.5 Model Training

Model training is a fundamental step in building machine learning systems. The dataset was applied to multiple models:

3.5.1 Single Models

Three distinct machine learning models were individually trained and evaluated. Hyperparameters were fine-tuned and optimized using Grid Search CV with multiple test sets; the best set was selected based on cross-validation to improve accuracy and reliability.

- Random Forest (RF): An ensemble of decision trees, where each tree is trained on randomly selected data, and the final prediction is determined by majority voting [13], [14], [15].

- XG Boost: It combines several weak learners sequentially to form a stronger predictive model, while correcting the errors of previous learners [16], [7].

- Support Vector Classifier (SVC): It creates optimal hyperplanes to separate different classes in the dataset, aiming to achieve maximum margin [17].

These models were selected for their proven effectiveness on tabular datasets, their ability to efficiently handle numerical and categorical features, and their complementary strengths. They are suitable as basic learners for stacking ensembles, while maintaining their computational executable and interpretability compared to more complex alternatives such as neural networks or graph-based models.

3.5.2   Stacking Model

Stacking This is an ensemble learning method where the results of a set of base models are added together. Base models get trained, and their prediction is utilized in a super-learner to come up with the final prediction. The primary benefit of it is that it makes use of the strengths of the various models to enhance the entire prediction accuracy [2]. The base learners, which were pre-trained in the current research, were Random Forest, XG Boost, and SVC, and the secondary learner, which was selected by the author in the current research, was Support Vector Machine (SVM). The secondary SVM model keeps the strongest aspects of the base models to yield a more accurate and robust prediction than when the base models are used individually [18], [6].

3.5.3   Cross-Validation Strategy

To optimize model hyperparameters and enhance generalization performance, a k-fold cross-validation strategy was employed using GridSearchCV. Specifically, 5-fold cross-validation (k = 5) was applied during the training phase, where the training data were partitioned into five equal folds. In each iteration, four folds were used for training and one fold was used for validation. This process was repeated five times, and the average validation performance was used to select the optimal hyperparameter configuration. The use of cross-validation helps reduce the risk of overfitting and ensures that the selected model generalizes well to unseen data.

3.6   Model evaluation

After the training and testing processes, the model's performance is measured through four metrics:

- Accuracy: The ratio of correct predictions to total predictions. [5].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad (1)$$

- Precision: The ratio of correctly predicted positive cases to all predicted positive cases [19].

$$Precision = \frac{TP}{TP + FP} \qquad (2)$$

- Recall: The ratio of correctly predicted positive cases to all actual positive cases [20].

$$Recall = \frac{TP}{TP + FN} \qquad (3)$$

- F- Measure: Harmonic mean of precision and recall, with the two measures balanced [21].

$$F - Measure = 2 * \frac{Precision * Recall}{Precision + Recall} \qquad (4)$$

Where: TP = True Positive, TN = True Negative, FP = False Positive, TP = True Positive.

The evaluation of the classifiers in this study focuses on accuracy, precision, recall, and F-measure, which are sufficient for assessing the performance of models on the balanced dataset used. While ROC-AUC is a popular metric for imbalanced datasets, in our case the dataset is balanced between real and fake accounts, making accuracy and F-measure more directly interpretable for this classification task. Hence, the omission of ROC-AUC does not affect the validity or comparability of the results."

4.   RESULTS AND DISCUSSION

Three single machine learning models, including the random forest (RF), XG Boost, and the SVC, were tested in the domain of fake Instagram account detection. Random Forest had the best performance of single models with an accuracy of 0.9421, precision of 0.9485, recall of 0.9361, and F-measure of 0.9423, as indicated in Table 2. The XG Boost bagged an accuracy of 0.9250, whilst SVC had the lowest accuracy of 0.8604. Adding a stacking ensemble with RF, XG Boost, and SVC as base learners and SVM as the meta-classifier further led to smaller results in which an accuracy of 0.9888, a precision of 0.9892, a recall of 0.9879, and an F-measure of 0.9886 are obtained (Table 2). The confusion matrices and performance indicators of each model are presented in Figures 5-8: Fig. 5 is the RF, Fig. 6 is the XG Boost, Fig. 7 is the SVC, and Fig. 8 is the stacking model with SVM as the last estimator. These numbers show that the stacking model is more predictive, and its performance lies in the merit of a number of base learners, decreased misclassification, and more solid detection of a fake account.

4.1 Statistical Significance Analysis:

To formally assess the reliability and statistical rigor of the observed performance improvements, a paired t-test was conducted on the 5-fold cross-validation accuracies of the stacking ensemble compared to each individual base model. The results of this analysis are as follows:

- Stacking vs. Random Forest: t = 0.0600, p = 0.9551 (not significant)

- Stacking vs. XG Boost: t = -1.1768, p = 0.3045 (not significant)

- Stacking vs. SVC: t = 53.4919, p < 0.01 (significant)

These results indicate that the stacking ensemble significantly outperforms the weaker base learner, SVC, confirming its effectiveness in combining models to correct the deficiencies of less accurate classifiers. However, the improvement over the stronger base learners (RF and XG Boost) is not statistically significant, which is expected since these models individually already provide high performance. This finding demonstrates that while the ensemble may not always dramatically outperform the best individual model, it provides stability, reduces variability, and ensures consistent high-level performance across different folds of data. Overall, this statistical validation strengthens the credibility of the proposed stacking ensemble and confirms that the observed performance gains are not due to random variation, highlighting the robustness of the model in detecting fake Instagram accounts.

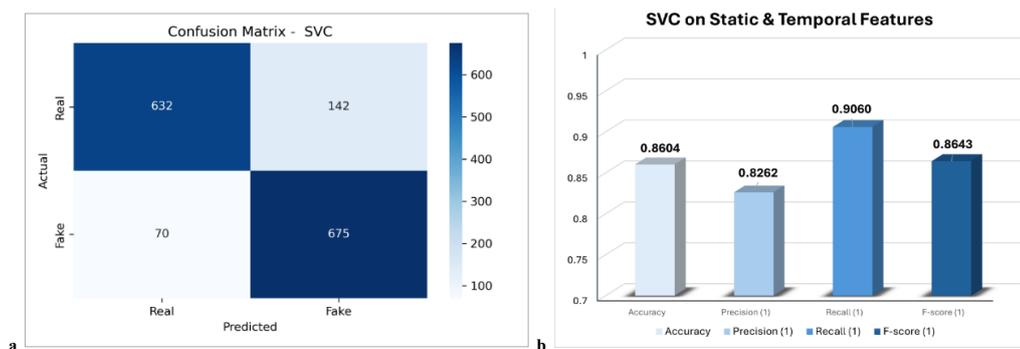**Table 2. - Performance of Single and Stacking Models**

| Model	Accuracy	Precision	Recall	F- Measure
Random Forest	0.9421	0.9485	0.9361	0.9423
XGBoost	0.9250	0.9071	0.9436	0.9250
SVC	0.8604	0.8262	0.9060	0.8643
Stacking Model	0.9888	0.9892	0.9879	0.9886
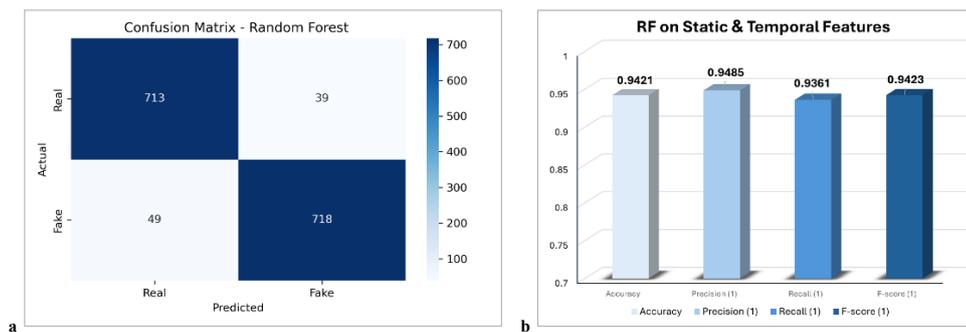
FIGURE 5. - RF (a) The Confusion Matrix; (b) The Performance Metrics
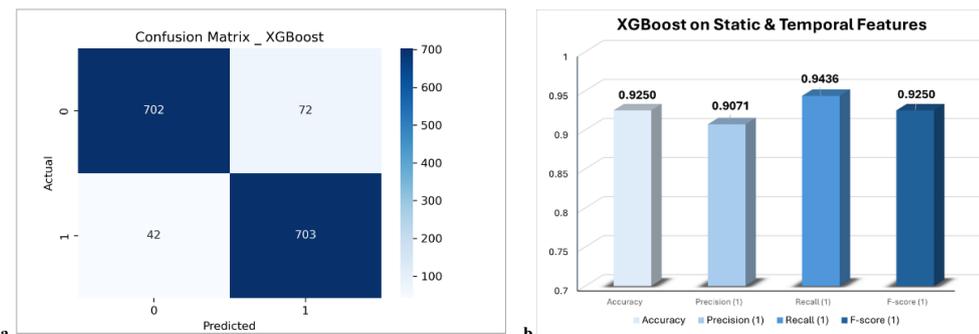
FIGURE 6. - XGBoost (a) The Confusion Matrix; (b) The Performance Metrics

FIGURE 7. - SVC (a) The Confusion Matrix; (b) The Performance Metrics
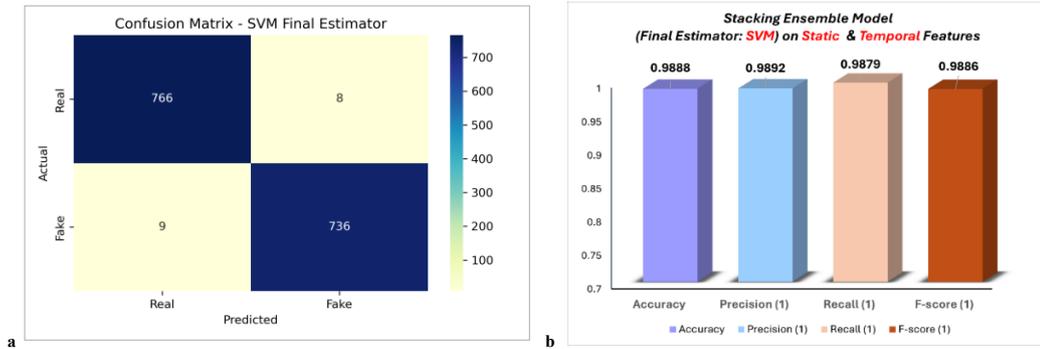
FIGURE 8. – Stacking– Final Estimator: SVM (a) The Confusion Matrix; (b) The Performance Metrics

5. COMPARATIVE ANALYSIS WITH PREVIOUS STUDIES

Table 3. - Comparison with Related Work

Ref	Methodology	Dataset / Source &Features (Count)	Accuracy (%)	Remarks
Proposed Work /2025	• Stacking (RF + XGB + SVC, final estimator SVM),	• 7,594 were collected through 4 methods, including manual • 17 features (Static11 + Temporal 5 and flag label)	• Stacking: 98.88 % • RF: 94.21 %	✓ It outperforms previous studies due to its large feature set, hybrid learner, and data collection. ✓ RF alone achieved high performance, outperforming many techniques in previous studies.

Ref	Author\Year	Methodology	Dataset / Source &Features (Count)	Accuracy (%)
1	A. Dey et al. /2019	LR, RF	696 / Kaggle, Only Static (11)	92.5%
2	F. C. Akyon, M. E. Kalfaoglu /2019	Naïve Bayes, LR, SVM, Neural Networks	Dataset1: 1,203 / manual classification based on account characteristics, Only Static (8) Dataset2: 1400 / Manual collection using Instagram API and bot behavior analysis from opensource tools over 6 months, Static (9) ,Temporal (8)	Dataset1: 94.0% Dataset2: 86.0%
3	Purba et al. /2020	RF, MLP, LR, Naïve Bayes, J48 DT	65,326 / The authentic users were taken from followers of 24 private university pages (8 Indonesian, 8 Malaysian, 8 Australian) on Instagram... The fake users were collected by buying followers from Indonesia sellers, from various sources, Static (16), Temporal (1)	90.09%
4	Keshav Kaushik /2022	Deep Neural Network, InstaFake, InstaReach	576 / Kaggle, Only Static (11)	91%
5	Sallah et al. /(2022)	RF, XGBoost, AdaBoost, SVM, DT, KNN	696 / Kaggle, Only Static (11)	96%

6	Alharbi et al. /2024	LSTM	Dataset1:1,203 /GitHub, Only Static (8)	Dataset1:97.42%
			Dataset2: 696 / Kaggle, Only Static (11)	Dataset2:94.21%
7	Chelas et al. /2024	RF, LR, MLP, KNN, DT, SVM, Gaussian Naïve Bayes	Dataset1: 1,194 /GitHub, Only Static (8) Dataset2: 696 / Kaggle, Only Static (11) Final Dataset (D1&D2): 1890, Only Static (8) features) + add 3 features	97.36%
8	B. G. Bokolo, Q. Liu / 2024	DT, LR, SVM, RF, KNN, XGBoost, Gradient Boosting, AdaBoost, Extra Trees	65,326 / Kaggle , Static (16) Temporal (1)	90%

6. CONCLUSION AND FUTURE WORK

This research proposed a new useful stacking ensemble algorithm to detect counterfeit Instagram accounts through the combination of static and dynamic attributes. The results of the experiment have shown that though all the individual models, including the Random Forest, XG Boost, and SVC, performed quite well to a certain extent, the Random Forest model among the single classifiers exhibited the best performance with high accuracy and an equal precision-recall measure. But most importantly, the stacking ensemble was the model that overcame all other models by having the highest accuracy, precision, recall and F-measure, which proved its best capability to define various patterns and reduce the occurrence of misclassification. The results indicate that the multiple classifiers in an ensemble structure are significant in utilizing the advantage of both base learners and improving the power of fake account detection systems. The incorporation of the temporal behavioral features also added to the enhancement of the reliability in the process of classification. To carry out the work further, a number of directions can be used to reinforce this research. The generalization that the model is able to perform will be enhanced by increasing the size of the dataset to cover more user behaviors. As well, more advanced methods of deep learning could allow identifying a broader range of fake account patterns or even more dynamic ones. The dynamic nature between behavioral traits and time can be further investigated to improve behavioral models, and the application of the model within a scalable and real-time system would facilitate its application to real-world situations of social media of large scale.

REFERENCES

[1] R. Sujitha and A. B, "Instagram fake and automated account detection: A review," 2023. [Online]. Available: https://doi.org/10.59544/dpaz6258/ngcesi23p29

[2] M. Schonlau, "Stacking," in *Springer Nature*, 2023, pp. 323–328. [Online]. Available: https://doi.org/10.1007/978-3-031-33390-3_15

[3] A. Dey, H. Reddy, M. Dey, and N. Sinha, "Detection of fake accounts in Instagram using machine learning," *Int. J. Comput. Sci. Inf. Technol.*, vol. 11, no. 5, pp. 83–90, 2019. [Online]. Available: https://doi.org/10.5121/ijcsit.2019.11507

[4] F. C. Akyon and E. Kalfaoglu, "Instagram Fake and Automated Account Detection," *arXiv preprint arXiv:1910.03090*, May 2022. [Online]. Available: https://arxiv.org/abs/1910.03090

[5] K. R. Purba, D. Asirvatham, and R. K. Murugesan, "Classification of Instagram fake users using supervised machine learning algorithms," *Int. J. Electr. Comput. Eng.*, vol. 10, no. 3, pp. 2763–2772, Jun. 2020. DOI: 10.11591/ijece.v10i3.pp2763-2772

[6] K. Kaushik, A. Bhardwaj, M. Kumar, S. Gupta, and A. Gupta, "A novel machine learning-based framework for detecting fake Instagram profiles," *Concurrency Computat.: Pract. Exper.*, Accepted. DOI: 10.1002/cpe.7349

[7] A. Sallah, E. A. A. Alaoui, S. Agoujil, and A. Nayyar, "Machine learning interpretability to detect fake accounts in Instagram," *Int. J. Inf. Secur. Privacy*, vol. 16, no. 1, pp. 1–24, 2022. DOI: 10.4018/IJISP.303665

[8] N. Alharbi, B. Alkalifah, G. Alqarawi, and M. A. Rassam, "Countering social media cybercrime using deep learning: Instagram fake accounts detection," *Future Internet*, vol. 16, no. 10, Art. no. 367, Oct. 2024. DOI: 10.3390/fi16100367

[9] S. Chelas, G. Routis, and I. Roussaki, "Detection of Fake Instagram Accounts via Machine Learning Techniques," *Computers*, vol. 13, no. 11, p. 296, Nov. 2024. [Online]. Available: https://doi.org/10.3390/computers13110296

[10] B. G. Bokolo and Q. Liu, "Advanced algorithmic approaches for scam profile detection on Instagram," *Electronics*, vol. 13, no. 8, Art. no. 1571, Apr. 2024. DOI: 10.3390/electronics13081571

[11] T. Makaba and E. M. Dogo, "A Comparison of Strategies for Missing Values in Data on Machine Learning Classification Algorithms," in *Proc. Int. Conf. Intell. Mechatronics Technol.*, 2019, pp. 1–7. DOI: 10.1109/IMITEC45504.2019.9015889

[12] "Categorical Data: Need, Encoding, Selection of Encoding Method and Its Emergence in Machine Learning Models—A Practical Review Study on Heart Disease Prediction Dataset Using Pearson Correlation," in *Lecture Notes Netw. Syst.*, 2023, pp. 369–382. DOI: 10.1007/978-981-19-6631-6_26

[13] J. Raju, "Machine Learning-Based Identification of False Profiles," *Indian Sci. J. Res. Eng. Manag.*, vol. 8, no. 4, pp. 1–5, 2024. DOI: 10.55041/ijsrem32257

[14] B. P. Kavin et al., "Machine learning-based secure data acquisition for fake accounts detection in future mobile communication networks," *Security Privacy*, vol. 2022, Art. no. 6356152. DOI: 10.1155/2022/6356152

[15] M. A. Qasim and N. A. Flayh, "Enhancing phishing website detection via feature selection in URL-based analysis," *Informatica*, vol. 47, no. 9, 2023. DOI: 10.31449/inf.v47i9.5177

[16] S. Madhu et al., "Profile Imposter Detection On Instagram Using XGBoost and SVM Algorithm," *Adv. Comput. Sci. Res.*, 2024, pp. 1453–1461. DOI: 10.2991/978-94-6463-471-6_141

[17] I. Bensassi, O. Ndama, M. Kouissi, and E. M. En-Naimi, "Enhancing fake profile detection through supervised and hybrid machine learning: a comparative analysis," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 37, no. 1, pp. 257–268, Jan. 2024. DOI: 10.11591/ijeecs.v37.i1.pp257-268

[18] M. A. Kausar, S. O. Fageeri, and A. Soosaimanickam, "Sentiment Classification based on Machine Learning Approaches in Amazon Product Reviews," *Eng. Technol. Appl. Sci. Res.*, vol. 13, no. 3, pp. 10849–10855, 2023. DOI: 10.48084/etasr.5854

[19] P. Manorom, U. Detthamrong, and W. Chansanam, "Comparative Assessment of Fraudulent Financial Transactions using the Machine Learning Algorithms Decision Tree, Logistic Regression, Naïve Bayes, K-Nearest Neighbor, and Random Forest," *Eng. Technol. Appl. Sci. Res.*, vol. 14, no. 4, pp. 15676–15680, 2024. DOI: 10.48084/etasr.7774

[20] E. Irwansyah, A. A. S. Gunawan, H. Pranoto, F. S. Pramudya, and L. Fakhriadi, "Deep learning with semantic segmentation approach for building rooftop mapping in urban irregular housing complexes," *Eng. Technol. Appl. Sci. Res.*, vol. 15, no. 2, pp. 20580–20587, 2025. DOI: 10.48084/etasr.9670

[21] A. A. Albishri and M. M. Dessouky, "A Comparative Analysis of Machine Learning Techniques for URL Phishing Detection," *Eng. Technol. Appl. Sci. Res.*, vol. 14, no. 6, pp. 18495–18501, 2024. DOI: 10.48084/etasr.8920