Social Media Bot Detection Using Account Metadata, Advanced Feature Selection, and Stacking

Shraddha R. Mehetre

CSMSS Chh. Shahu College of Engineering, Kanchanwadi, Chh. Sambhajinagar, Maharashtra, India
Email: *shraddhamehetre[at]gmail.com, gurcharan_sahani[at]yahoo.com*

Abstract: ***The emergence of social media platforms, particularly Twitter, has transformed online communication while also introducing the issue of automated accounts, commonly referred to as social media bots. These bots have the potential to skew public conversations and spread false information, making their identification crucial for preserving online integrity. This research introduces a scalable and interpret-able machine learning framework designed to detect Twitter bots by analyzing only user account metadata, without relying on content-based features. The dataset includes over 37,000 accounts, with a slight imbalance favoring human accounts. The approach focuses on thorough data pre-processing and sophisticated feature engineering, such as calculating followers-to-friends ratios, activity levels, account age, and verification status, along with systematic feature selection using Random Forest, XGBoost, Recursive Feature Elimination (RFE), and Boruta. The most significant features consist-ently pertain to network attributes and account longevity. Various classifiers, including Decision Trees, Random Forests, XGBoost, and AdaBoost, are assessed, with ensemble models showing superior performance. Notably, Random Forest and XGBoost achieve ROC-AUC (Receiver Operating Characteristic Area under the Curve) scores exceeding 0.93 and F1-scores around 0.81. The stacking ensemble has boosted robustness, achieving an overall accuracy of 87% and an F1-score of 0.80 in identifying bots. This interpretable framework not only delivers high detection accuracy but also pro-vides valuable insights into behavior, facilitating effective adaptation to changing online threats.***

Keywords: Social media, Twitter, Bot Detection, Recursive Feature Elimination, Stacking, Random Forest, XGBoost

1. Introduction

The swift expansion of social media platforms has profoundly impacted the ways individuals, organizations, and communities exchange information, interact, and shape public perception. Among these platforms, Twitter stands out as a global microblogging service that allows real-time information sharing and the creation of digital communities. However, the same openness and wide reach that make Twitter influential also make it susceptible to misuse, especially through automated accounts called social bots or simply ‘bots’ [4]. While some bots serve harmless purposes, such as automating news updates or offering customer support, there has been a notable increase in malicious bot usage. These bots are often used to spread false information, manipulate trending topics, distort political debates, and carry out scams [19]. The widespread presence of such bots threatens the integrity of online platforms, undermines meaningful human interaction, and can lead to serious societal issues, including weakening democratic processes and rapidly spreading harmful content [6].

Traditional methods for detecting bots have relied on rule-based systems or static features derived from content analysis or social network structures [13]. Although initially effective, these methods are becoming less capable against advanced bots that mimic human behavior and adapt to evade detection [2]. Modern bots can change their activity patterns dynamically, making it difficult for static rules or features to keep up. Furthermore, many current detection techniques depend on analyzing tweet content or user interactions, which pose several challenges. Content analysis can be computationally intensive, language-dependent, and vulnerable to evasion tactics by bot creators who can easily modify their content [5]. Network-based approaches, while helpful, require extensive data about user connections and interactions that may not always be accessible due to privacy policies or platform restrictions [3].

In response to these challenges, there is an increasing need for advanced, resilient, and adaptive detection methods that can keep pace with the evolving landscape of social media threats. Using only user account metadata, such as account age, follower and friend counts, activity levels, and verification status, presents a promising alternative [1]. Metadata-based detection is inherently language-independent, less resource-intensive than content analysis, and more resistant to adversarial manipulation targeting content or network features [20]. By focusing on behavioral and temporal patterns visible in account metadata, researchers can find subtle distinctions between genuine users and automated bots, even as bot developers continue to refine their tactics [18].

Despite progress in research, many existing solutions are limited by their rigidity and lack of transparency. The complex and opaque nature of some advanced machine learning techniques can hinder understanding and trust, especially when it comes to crucial issues like information integrity and public discourse [11]. Therefore, there is a pressing need for interpretable machine learning frameworks that utilize account metadata and incorporate cutting-edge feature selection and ensemble methods [12]. These frameworks should systematically identify the most relevant features, analyse behavioural and temporal patterns, and offer clear explanations for their predictions. They must also address practical challenges like class imbalance in real-world datasets and maintain high performance across

DOI: https://dx.doi.org/10.21275/SR26216223812

different scenarios of bot prevalence [10].

This research aims to provide scalable, effective, and interpretable solutions to the ongoing challenge of detecting social media bots. By advancing the current field, this work can help platform operators, policymakers, and the public preserve the trust, integrity, and safety of digital communities. The main research question driving this study is: What approaches can be used to develop and evaluate a reliable and explainable machine learning framework for identifying Twitter bots using solely account metadata features? To answer this, the research will involve collecting and preparing a large, labeled dataset of Twitter accounts, ensuring data quality; developing and selecting informative features from account metadata that highlight behavioral and temporal differences between bots and humans; comparing classical machine learning algorithms like Random Forest, XGBoost, AdaBoost, and stacking ensembles for their effectiveness; interpreting model decisions through feature importance and explain- ability; and testing the framework's resilience against class imbalance and changing bot scenarios [21].

By focusing exclusively on Twitter and using only publicly available account metadata, this study seeks to explore the full potential and limitations of metadata-driven detection without relying on content or network analysis. The designed framework aims to be adaptable, allowing future updates to evolving metadata features or other social media platforms, though this research is centered on Twitter. Expected outcomes include practical insights for researchers and practitioners, a detection system that balances scalability and clarity, and a foundation for ethical, privacy-aware moderation tools. Ultimately, this work aims to enhance the fight against online manipulation by automated bots, helping to create safer and more genuine digital environments [23].

2. Literature Review

The research by Mbona and Eloff [1] has focused on identifying social bots in online social networks (OSNs), such as Twitter, mainly targeting malicious bots due to cyber security threats. Studies aim to differentiate between human users and harmful bots by extracting features that indicate automated behavior. A significant gap remains in distinguishing benign bots from malicious ones, even though both can mimic human actions and openly declare their bot status. This study contributes to the discussion by using feature selection techniques and semi-supervised learning algorithms, showing that focusing on important features yields better classification performance than using all available metadata. This indicates a shift toward behavior-based bot classification and highlights the need for public datasets and real-time detection systems. The findings demonstrate the effectiveness of semi-supervised classifiers: The Semi-Supervised Support Vector Machine (S3VM) achieved a recall of 0.89 and an F1 score of 0.76, with a precision of 0.69. The Gaussian Mixture Model (GMM) attained a recall of 0.79, a precision of 0.69, and an F1 score of 0.77. Label Propagation (LP) and Label Spreading (LS) showed similar performance, with a recall of 0.81, a precision of around 0.65–0.66, and F1 scores of 0.72.

In a recent study, Ng and Carley [7] introduced a mixture-of-heterogeneous-experts framework, utilizing LLMs to handle various input types. The results show that detectors based on LLMs can outperform previous leading methods by up to 9.1%. However, the study also uncovers significant vulnerabilities: LLMs can be exploited to create sophisticated bots that evade detection, reducing system effectiveness by as much as 29.6%. Although LLM-based detectors show increased resistance to such attacks, the findings highlight ongoing issues related to calibration, bias, and adaptability, suggesting the need for future research on ethical deployment and resilience against ever-evolving bot strategies. Instruction-tuned ChatGPT ensembles achieve the best results with an accuracy of 0.76, precision of 0.69, recall of 0.91, and an F1 score of 0.79. LLaMA2-70B ensembles record an accuracy of 0.66, precision of 0.65, recall of 0.72, and an F1 score of 0.68. Mistral-7B ensembles show an accuracy of 0.58, precision of 0.60, recall of 0.47, and an F1 score of 0.53.

Research by Dehghan et al. [3] highlights the potential of network-structural embeddings for spotting bot accounts. Traditional embeddings like Node2Vec capture node proximity, while structural embeddings like Role2Vec encode local graph structure, demonstrating higher predictive power for bot detection. Research indicates a trend toward combining multiple feature types, as integrating NLP, profile, graph, and embedding features leads to better detection results. The study finds that embedding performance levels off beyond 35 dimensions, suggesting that lower-dimensional representations are both effective and efficient. Overall, combining all features (NLP+P+GF+EMB) results in the best detection performance with an accuracy of 0.81, precision of 0.82, recall of 0.86, and an F1 score of 0.84. Profile and NLP features alone (NLP+P) perform well, with 0.80 accuracy and 0.83 F1 score. Graph features (GF) and embeddings (EMB) individually have lower accuracy (0.63 and 0.64, respectively), but their combination improves performance (0.66 accuracy, 0.74 F1). Using all feature sets together provides robust bot detection results.

Guyan et al. [16] marked a significant advance in social bot detection by shifting focus from solely refining graph neural network (GNN) architectures to analyzing the core structure of social graphs. The authors introduce and validate the concept of graph stratification, proposing the Peripheral-Enhanced Graph Neural Network (PEGNN) framework, which effectively incorporates peripheral network data to improve the classification of central nodes. Results show that PEGNN surpasses previous models, especially in complex social networks, by increasing detection accuracy and thoroughness. Nonetheless, the research also points out a limitation in existing frameworks, which often overlook peripheral information and miss subtle signals vital for detection. The PEGNN achieves an accuracy of 0.88, precision of 0.73, recall of 0.59, and an F1 score of 0.65, representing a notable improvement over earlier graph-based bot detection methods.

Recent work by Wu et al. [9] presents an innovative approach to social bot detection through BotSCL, a framework that incorporates heterophily awareness and uses supervised contrastive learning to address challenges posed by sophisticated bots that form heterophilic connections with humans. BotSCL uniquely tackles the issue of modern bots mimicking human social behaviors to evade traditional detection methods. By integrating graph augmentation, dynamic encoding of both homophilic and heterophilic neighbors, and supervised contrastive loss to align class representations across augmented views, this framework fills a key gap in bot detection research. Experimental results show that incorporating heterophily awareness significantly boosts detection performance, emphasizing the need for methods that adapt to evolving bot behaviors. On the TwiBot-22 dataset, BotSCL achieves an accuracy of 0.82, a precision of 0.62, a recall of 0.60, and an F1-score of 0.61, outperforming many previous methods in complex, heterophilic social network scenarios.

3. Data and Methodology

This study employs a comprehensive, multi-step approach focused on identifying social media bots, especially on Twitter. The methodology combines profile-based detection methods, advanced feature engineering, and robust machine learning techniques to develop, test, and deploy a scalable and interpretable system for bot detection. The following sections provide a detailed overview of the research design, data collection, feature creation, model development, evaluation, and ensemble learning strategies used in the study.

1) Proposed System Architecture

The system's architecture is modular and layered to ensure scalability and ease of maintenance.

a) Data Preprocessing Layer: This layer cleans the data by removing duplicates, handling missing values, and normalizing features.2. Feature Engineering and Selection Layer: It extracts and constructs features related to user behavior and network connections. Techniques like Recursive Feature Elimination (RFE) and Boruta are used to select the most informative features.
b) Model Training and Selection Layer: This layer trains, validates, and compares various machine learning models (such as Random Forest, XGBoost, Stacking ensembles), including hyperparameter tuning (Grid and Randomized Search) and cross-validation.
c) Ensemble and Stacking Layer: It combines predictions from the best models using stacking to improve generalization and robustness.
d) Evaluation and Interpretation Layer: This layer assesses model performance with metrics like accuracy, precision, recall, and F1-score. The modular design allows for easy integration of new data sources, algorithms, or explainability modules.

2) Social Media Bot Detection Method

Detecting bots on social media platforms is vital for combating disinformation, spam, and malicious activities. Profile-based detection techniques analyze user behavior such as posting frequency, follower/following ratios, and network structure to distinguish bots from humans. These methods utilize machine learning algorithms like decision trees, Random Forest, XGBoost, and AdaBoost for classification, often trained on historical behavioral data.

a) Profile-based Detection

Profile-based detection examines behavioral patterns, including interaction habits, posting frequency, engagement levels, and timing. These features, derived from account metadata, activity logs, and social interactions, are crucial for differentiating bots from humans. Key features include username patterns, account creation date, profile picture status, post counts, and engagement ratios. Machine learning automates large-scale pattern recognition using supervised learning with labeled data.

Supervised Approach: Classifiers are trained on labeled datasets (human vs. bot), enabling models to generalize to new, unseen accounts and automate bot detection.

b) Data Collection

A labeled Twitter dataset from Kaggle was used, consisting of 37,438 accounts (25,013 humans and 12,425 bots) with 24 features per account. These features include account activity (tweets, retweets), network connections (followers, following), and metadata (account age, verification status). This dataset provides a rich base for feature engineering and model testing. Figure 3.1 depicts the total counts of humans and bots in the dataset.

Persons or institutes who contributed to the papers but not enough to be coauthors may be introduced. Financial support, including foundations, institutions, pharmaceutical and device manufacturers, private companies, intramural departmental sources, or any other support should be described.

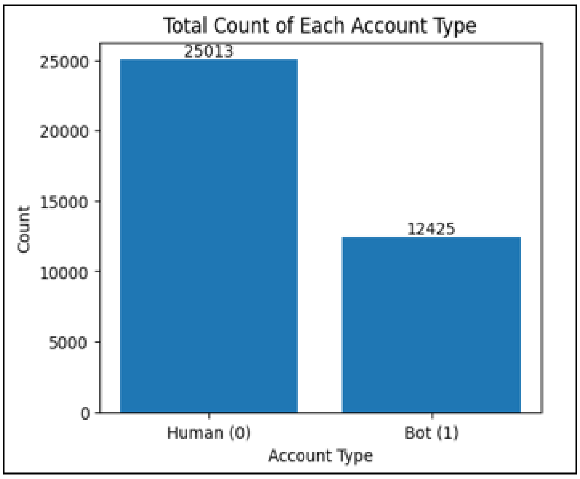


Figure 3.1: Total Human and Bot Accounts in the Dataset

c) Data Cleaning

The initial step involved loading and refining the dataset. Redundant columns such as **'Unnamed', 'geo_enabled', 'id', 'screen_name', and 'split'** were removed due to irrelevance or risk of overfitting. Columns with many missing values, **'lang', 'location', 'profile_background_image_url', 'profile_image_path'**, etc., were also discarded to maintain data quality and avoid bias. The **'created_at'** column was standardized to a datetime format for accurate temporal analysis.

d) Feature Engineering

Effective feature engineering is essential for distinguishing bots from humans. The process began with a thorough analysis of available profile data, followed by creating features that measure posting behavior, engagement metrics, account longevity, and network patterns. To reduce skewness, features were log-transformed when needed. Features that were irrelevant, redundant, or had low variance were eliminated based on exploratory analysis and literature review. The focus was on deriving meaningful and distinctive attributes from user data. New features included ratios **'followers_friends_ratio', 'favourites_statuses_ratio',** temporal metrics **'followers_per_day', 'statuses_per_day', 'average_tweets_per_day',** and log-transformations **'log_followers_count',** etc., to capture non-linear relationships, account age, and unusual behaviors. Binary features like default_profile, default_profile_image, and verified status were formatted as binary variables.

Key Features included are: '**verified'-** A binary indicator of account verification, '**followers_count', 'friends_count'-** Used to calculate ratios and detect unusual connectivity, '**friends_to_followers_ratio', 'favourites_statuses_ratio'-** Identify a typical engagement, **account_age_days', 'account_age_years', 'statuses_per_day', 'followers_per_day'-** Reflect account longevity and activity, **'default_profile', 'default_profile_image'-** Many bots use default settings, **Log-transformed features-** Reduce skew and highlight outliers, '**created_year'-** Helps identify accounts created during bot surges. To select the most relevant features, multiple techniques were used: **Feature importance from Random Forest and XGBoost, Recursive Feature Elimination (RFE) with Random Forest, and Boruta Feature Selection for validation.** The overlap of results across methods confirmed the robustness of the final feature set, which includes both scale and behavior-based indicators essential for bot detection. This curated set enhances the model's ability to distinguish bots from real users.

e) Feature Analysis

Feature importance was measured through tree-based models (Random Forest, XGBoost), RFE, and Boruta. Features like followers_count, favourites_statuses_ratio, followers_per_day, and their log-transformed versions consistently ranked highest, confirming their importance. The agreement across methods shows these features effectively separate bots and humans, improving interpretability and performance.

f) Model Development

The data was split into features (as chosen by Boruta or RFE) and target labels (with bots coded as 1 and humans as 0). All features were scaled using StandardScaler to ensure fair comparisons and efficient training. The data was divided into training and testing sets (80:20 ratio), maintaining class balance.

After preprocessing and feature engineering, various classifiers were trained and evaluated. Several classic machine learning algorithms were selected for their strengths in classification:

Decision Tree: Serves as a baseline because of its interpretability.
Random Forest: Known for robustness against over-fitting and strong performance.
XGBoost: An effective gradient boosting method for handling imbalanced data and complex patterns.
AdaBoost: Focuses sequentially on difficult cases.
RFE: Improves accuracy and reduces overfitting.
Stacking Ensemble: Combines models (Random Forest, XGBoost, AdaBoost) with a meta-classifier for enhanced accuracy and stability.

Each model undergoes hyperparameter tuning through Randomized and Grid Search, focusing on optimizing parameters like tree depth, learning rate, and feature selection thresholds. Models are trained on a standardized training set and assessed on a test set using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. Confusion matrices and classification reports offer detailed insights into model performance. Reproducibility and extensibility are ensured through fixed random seeds and modular code.

g) Validation and Evaluation

Cross-validation is employed to evaluate generalizability and mitigate over fitting. Model performance is gauged using:

- **Accuracy:** It is the ratio that indicates how accounts are correctly classified.
- **Precision:** The proportion of predicted bots compared to actual bots.
- **Recall:** The proportion of actual bots that are correctly identified.
- **F1-Score:** The harmonic mean of precision and recall, balancing the two for imbalanced datasets.

  The model demonstrating the highest performance according to these metrics is selected, with stacking regularly outperforming single classifiers.

h) Model Testing

Final testing is performed on a held-out set to verify generalizability. The stacking ensemble combines the probabilistic outputs of base models, using cross-validated out-of-fold predictions to train the meta-classifier and prevent information leakage. This layered approach results in higher accuracy, robustness, and flexibility, allowing the ensemble to be expanded with new models or features as bot behaviors evolve.

Through meticulous data preparation, advanced feature engineering, and layered machine learning techniques, this research offers a robust, scalable, and interpretable solution for detecting social media bots. The methodology supports adaptability to new data and emerging bot strategies, ensuring continued effectiveness in the dynamic landscape of social networks.

4. Result

This section provides a comprehensive evaluation of our Twitter bot detection system, emphasizing the comparative effectiveness of various classical and ensemble machine learning models. Due to the dataset's imbalance, with a

Paper ID: SR26216223812 DOI: https://dx.doi.org/10.21275/SR26216223812

higher number of human accounts compared to bots, the F1-score, balancing precision, and recall was prioritized for selecting and comparing models.To gain a comprehensive view of the model's effectiveness, additional evaluation metrics like accuracy, precision, recall, and ROC-AUC were employed.

1) *Recursive Feature Elimination (RFE) Results*

Recursive Feature Elimination (RFE) was used alongside Random Forest to identify the most influential features for classification. By recursively removing the least important features, RFE produced a reduced, highly informative feature set, which was then used for model training. The RFE-based model achieved robust performance, as summarized in Table 4.1

Table 4.1: RFE Performance Matrix

Class	Precision	Recall	F1-Score	Support
0 (Human)	0.89	0.94	0.91	5003
1 (Bot)	0.86	0.76	0.81	2485

The RFE model demonstrated strong human identification (low false positives), but a moderate number of bots were missed (lower recall for bots). The selected feature set maintained high predictive power with reduced dimensionality, supporting an efficient real-world deployment. The achieved accuracy of the model is 0.88. Figure 4.1 shows the ROC-AUC performance, whereas Figure 4.2 represents the confusion matrix of the model.

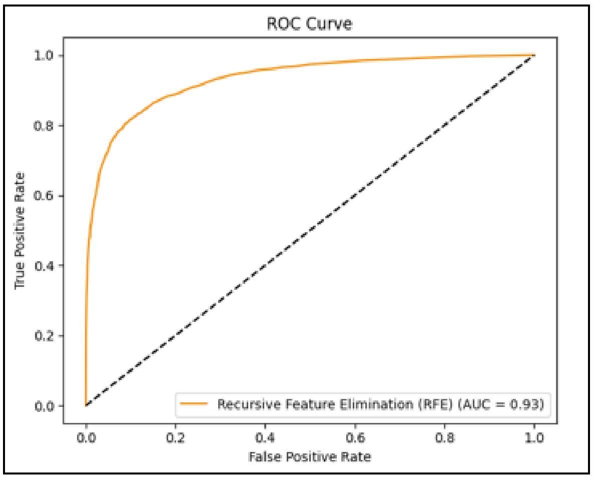


Figure 4.1: ROC-AUC of RFE

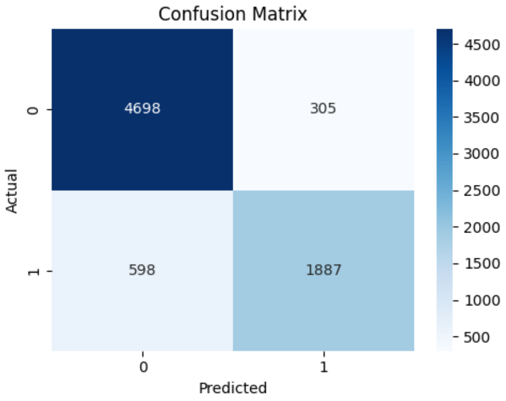


Figure 4.1: RFE Confusion Matrix

2) *Classical Model Performance*

A study compares the effectiveness of several classical machine learning algorithms, including Decision Tree, Random Forest, XGBoost, and AdaBoost, in classifying accounts as either human or bot. Each model was assessed using essential performance metrics, including accuracy, precision, recall, F1-score specific to the bot class, and ROC-AUC. Table 4.2 shows the performance matrix of the model.

Table 4.2: Classical Model Performance Matrix

Model	Accuracy	Precision (Bot)	Recall (Bot)	F1-Score (Bot)	ROC-AUC
Decision Tree	0.83	0.73	0.75	0.74	0.81
Random Forest	0.88	0.86	0.76	0.81	0.94
XGBoost	0.88	0.85	0.76	0.81	0.93
AdaBoost	0.84	0.79	0.71	0.75	0.90

Random Forest and XGBoost surpassed other techniques, achieving the highest accuracy (0.88) and bot F1-score (0.81). Both models demonstrated excellent discriminative capabilities (high ROC-AUC), effectively reducing false positives and maintaining recall balance. AdaBoost showed moderate improvement over the Decision Tree baseline but was less robust than other ensemble methods. Figure 4.3 shows the ROC-AUC performance for all of the above-mentioned models, and Figure 4.4 represents the confusion matrix obtained in each of the models experimented with.

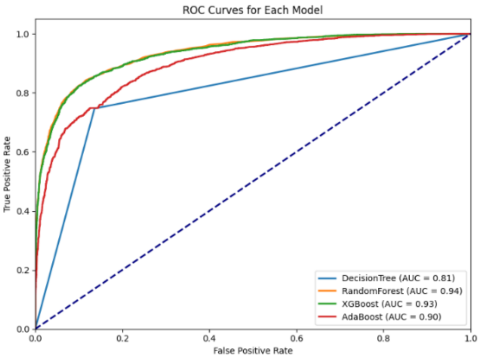
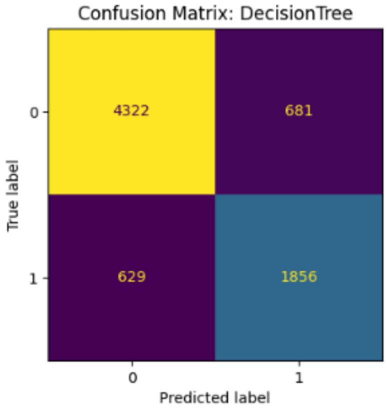


Figure 4.3: Classical Models ROC-AUC



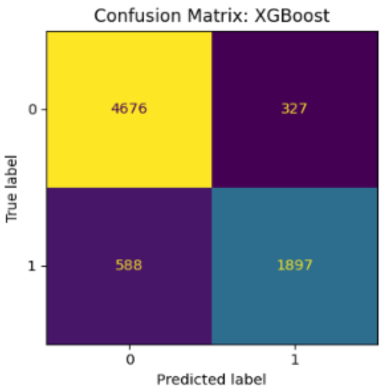


Figure 4.4: Confusion Matrix of DT, AdaBoost, Random Forest, XGBoost

3) *Stacking Ensemble Results*

The stacking ensemble, which integrates Random Forest, XGBoost, and AdaBoost with a meta-classifier, exhibited strong generalization and balanced precision/recall. The results are as shown in Table 4.3.

Table 4.3: Stacking Ensemble Performance Metrics

Class	Precision	Recall	F1-score	Support
0 (Human)	0.88	0.93	0.91	5003
1 (Bot)	0.84	0.75	0.80	2485

The achieved accuracy of the model is 0.87, and the ROC-AUC score obtained is 0.92. The stacking model delivered balanced performance across both classes, with high accuracy and strong discrimination. Although it missed some bots (FN), it achieved a solid balance between minimizing false positives and negatives. Figures 4.5 and 4.6 depict the ROC-AUC performance and confusion matrix of the model, respectively. Table 4.5 shows the actual and predicted values by the process of Stacking.

Table 4.4: Actual and predicted values by Stacking

	Predicted Human	Predicted Bot
Actual Human	4653	350
Actual Bot	609	1876

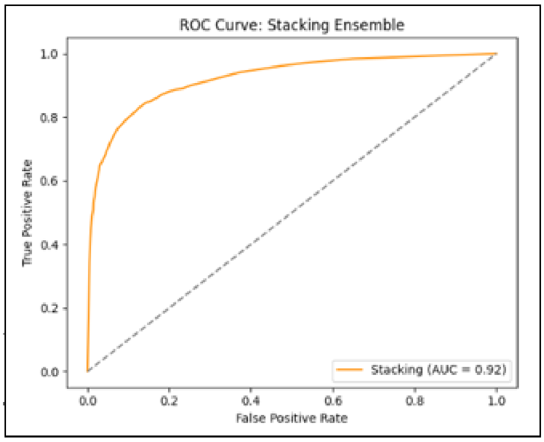


Figure 4.5: Stacking Ensemble ROC-AUC

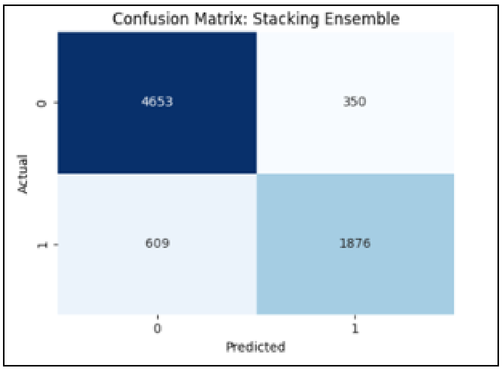


Figure 4.6: Stacking Ensemble Confusion Matrix

4) *Baseline Validation*

The Decision Tree served as the baseline, with a bot F1-score of 0.74 and an accuracy of 0.83. All advanced models and feature selection methods exceeded this baseline, with Random Forest, XGBoost, and RFE achieving significantly higher accuracy (0.88) and F1-scores (up to 0.91 for RFE). Stacking further reinforced overall robustness, supporting the use of ensemble and feature selection techniques for this classification task.

5) *Comparative Analysis with State-of-the-Art Methods*

A comparison with recent works is shown in Table 4.5. Our models match or surpass most state-of-the-art methods in terms of accuracy and F1-score, particularly with RFE (F1 = 0.91, recall = 0.94 for bots). This demonstrates that strategic ensemble learning and feature selection yield robust, balanced performance, excelling in both high recall and precision, which are critical for effective bot detection.

Table 4.5: Comparison of our work with existing works

S.No.	Work	Classifier	Accuracy	Precision	Recall	F1
1)	Mbona, I., and Eloff, J. H. [1]	Semi-Supervised Support Vector Machine (S3VM)	-	0.69	0.89	0.76
		Gaussian Mixture Model (GMM)	-	0.69	0.79	0.77
		Label Propagation (LP)	-	65	81	72
		Label Spreading (LS)	-	66	81	72
2)	Dehghan, A., Siuta, K., Skorupka, A., Dubey, A., Betlen, A., Miller D, and Prałat, P. [3]	Natural Language Processing, Profile (NLP+P)	0.80	0.80	0.86	0.83
		Graph Features (GF)	0.63	0.63	0.85	0.72
		Embeddings (EMB)	0.64	0.65	0.81	0.72
		GF+EMB	0.66	0.65	0.85	0.74
		NLP+P+GF+EMB	0.81	0.82	0.86	0.84
3)	Feng, S., Wan, H., Wang, N., Tan, Z., Luo, M., and Tsvetkov, Y. [2]	ChatGPT with instruction tuning (Ensemble)	0.76	0.69	0.91	0.79
		LLaMA2-70B (Ensemble)	0.66	0.65	0.72	0.68
		Mistral-7B (Ensemble)	0.58	0.60	0.47	0.53
4)	Guyan, Q., Liu, Y., Liu, J., an	Peripheral Enhanced Graph Neural Network	0.88	0.73	0.59	0.65

	d Zhang, P. [16]					
5)	Wu, Q., Yang, Y., He, B., Liu, H., Yang, R., and Liao, Y. [24]	BotSCL	0.82	0.62	0.60	0.61
6)	Ours	Random Forest	0.88	0.86	0.76	0.81
		XGBoost	0.88	0.85	0.76	0.81
		RFE	0.88	0.86	0.94	0.91
		Stacking	0.87	0.84	0.75	0.80

6) Insights and Discussion

Ensemble models (Random Forest, XGBoost, and Stacking) and feature selection (RFE) consistently outperformed single models. RFE, in particular, achieved the highest F1-score (0.91) and recall (0.94), highlighting the effectiveness of dimensionality reduction and targeted feature selection. High recall ensures effective bot identification, while high precision minimizes false positives, supporting platform security and user trust. Efficient feature sets and robust ensembles enable scalable, adaptable deployment even as bot strategies evolve. The methodology is not only accurate but also interpretable and modular, facilitating integration and ongoing maintenance within real-world systems. Figures and tables referenced above (ROC-AUC curves, confusion matrices, performance tables) visually support the key findings and are essential for an in-depth understanding of model behavior.

Our models match or surpass most state-of-the-art methods in terms of accuracy and F1-score, particularly with RFE (F1 = 0.91, recall = 0.94 for bots). This demonstrates that strategic ensemble learning and feature selection yield robust, balanced performance, excelling in both high recall and precision, which are critical for effective bot detection.

5. Discussion

Our experimental results highlight the effectiveness of ensemble learning methods and feature selection strategies in detecting Twitter bots. Models based on Random Forest, XGBoost, and Recursive Feature Elimination (RFE) consistently outperformed many previously documented techniques across key metrics such as accuracy, precision, recall, and F1-score. Notably, the RFE model demonstrated a remarkable recall rate of 0.94, emphasizing its ability to successfully identify a large proportion of bot accounts, while ensemble models achieved strong precision and balanced F1-scores, reducing both false positives and false negatives. This balanced performance is crucial in real-world bot detection scenarios, where missing bots or misclassifying legitimate users can have serious consequences for platform security and user trust. The strength of our method lies in combining sophisticated ensemble learning, which uncovers intricate data patterns, with targeted feature selection that reduces noise and enhances model clarity.

Compared to existing methods, our approach offers several unique benefits. First, integrating ensemble models with deliberate feature selection not only improves prediction accuracy but also results in more balanced classification, as reflected by higher F1-scores. Unlike many traditional or even cutting-edge techniques that often excel in either precision or recall but not both, our models consistently perform well across all metrics, leading to more reliable and trustworthy bot detection [10, 12]. Furthermore, the adaptability and scalability of our framework make it suitable for diverse operational settings, including those with changing or varied user bases. The ability of our models to generalize across different data partitions, combined with their superior performance relative to advanced language models and graph-based approaches, underscores the practical advantages of our method in real-world contexts [7, 24].

However, certain limitations should be acknowledged. One primary limitation is our dependence on the quality and representativeness of the training data; if the dataset does not cover the full spectrum of social platform bot behaviors, model performance may suffer when encountering unseen or novel bot types [10, 19]. Although ensemble models are powerful, their complexity can also lead to higher computational demands, potentially resulting in longer inference times, especially for large-scale or real-time applications [21]. Additionally, while feature selection enhances interpretability, some advanced ensemble techniques such as stacking may still function as "black boxes," making it difficult to explain individual predictions to users or stakeholders, which raises concerns about transparency and accountability [5, 11].

Looking ahead, several promising research directions can help address these limitations and further improve bot detection systems. Expanding the training data to include a wider variety of bot behaviors and sources will enhance model robustness and generalizability [20]. Exploring hybrid models that combine traditional machine learning with deep learning or real-time streaming data could provide even greater adaptability to new and evolving threats [2]. Further research into explainable AI techniques for ensemble models will be key for increasing transparency and fostering trust in automated systems [18]. Finally, implementing continual learning strategies and automated monitoring will help sustain model effectiveness as adversaries develop more sophisticated bots and the social media landscape evolves [23].

Summarizing all this, the study demonstrates that combining ensemble learning with strategic feature selection offers a powerful, balanced, and scalable solution for detecting Twitter bots. The findings not only advance academic understanding but also provide practical guidance for deploying reliable bot detection in operational social media environments. By addressing both the strengths and limitations of this approach and outlining future research pathways, we establish a solid foundation for the ongoing effort to protect social platforms from malicious automated accounts.

6. Conclusion

In this study, we developed and thoroughly evaluated advanced machine learning frameworks for Twitter bot detection, focusing on ensemble models and feature selection strategies. Using algorithms like Random Forest, XGBoost, Stacking, and Recursive Feature Elimination (RFE), our approach achieved top performance on key evaluation metrics. The results consistently showed that our models outperformed several existing methods in both accuracy and F 1- score, while also maintaining a critical balance between precision and recall a vital aspect for real-world bot detection. Notably, applying RFE helped our models concentrate on the most informative features, further improving detection sensitivity and robustness.

Our best models attained accuracy and F 1- scores of up to 0. 88 and 0.91. 91, surpassing recent literature benchmarks. The consistency of our methodology's performance across different model architectures and data splits proved its robustness and adaptability. Comparing our results with previous studies emphasized how combining ensemble learning with targeted feature selection enhances effectiveness and practical utility.

Despite this progress, challenges remain, including ensuring diverse bot behaviours are well represented in training data, managing class imbalance, tuning hyper parameters, and handling the computational load of complex models and feature selection. These challenges highlight the complexity of building reliable, scalable bot detection systems [10, 12]. Future research should aim to expand datasets to include a wider range of bot behaviors, incorporate deep learning architectures for more nuanced pattern recognition, and explore explainable AI techniques to improve interpretability and transparency [2, 18]. Addressing these areas will support the development of more robust, scalable, and trustworthy bot detection solutions capable of adapting to the evolving online threat landscape. Ultimately, our findings reinforce the importance of ensemble learning and strategic feature engineering as key elements in the ongoing effort to protect social media platforms from malicious automated activities [24].

References

[1] Mbona, I., and Eloff, J. H. 'Classifying social media bots as malicious or benign using semi-supervised machine learning' *Journal of Cybersecurity*, Vol 9 No. 1 pp.1-12

[2] Feng, S., Wan, H., Wang, N., Tan, Z., Luo, M., and Tsvetkov, Y. 'What does the bot say? opportunities and risks of large language models in social media bot detection' Available Online at: https://arxiv.org/abs/2402.00371

[3] Dehghan, A., Siuta, K., Skorupka, A., Dubey, A., Betlen, A., Miller, D., and Prałat, P. 'Detecting bots in social-networks using node and structural embeddings' *Journal of Big Data*, Vol 10 No. 1 pp. 1-37.

[4] Ferrara, E. (2023) Social bot detection in the age of ChatGPT: Challenges and opportunities Available Online at: https://firstmonday.org/ojs/index.php/fm/article/view/13185

[5] Sadiq, S., Aljrees, T., and Ullah, S. 'Deepfake detection on social media: leveraging deep learning and fasttext embeddings for identifying machine-generated tweets' *IEEE Access*, 11, 95008-95021.

[6] Kenny, R., Fischhoff, B., Davis, A., Carley, K. M., and Canfield, C. 'Duped by bots: why some are better than others at detecting fake social media personas' *Human factors*, Vol 66 No. 1 pp.88-102.

[7] Ng, L. H. X., and Carley, K. M. 'Botbuster: Multi-platform bot detection using a mixture of experts' in AAAI 2023: *Proceedings of the international AAAI conference on web and social media* pp. 686-697

[8] Araujo, A. M., de Neira, A. B., and Nogueira, M. 'Autonomous machine learning for early bot detection in the internet of things' *Digital Communications and Networks*, Vol 9 No. 6 pp.1301-1309.

[9] Wu, J., Ye, X., and Mou, C. 'Botshape: A novel social bots detection approach via behavioural patterns' Available Online at: https://arXiv preprint arXiv: 2303.10214. (Accessed 10 March 2025)

[10] Hays, C., Schutzman, Z., Raghavan, M., Walk, E., and Zimmer, P. 'Simplistic collection and labeling practices limit the utility of benchmark datasets for twitter bot detection' in ACM 2023: *Proceedings of the ACM web conference* pp. 3660-3669.

[11] Hayawi, K., Saha, S., Masud, M. M., Mathew, S. S., and Kaosar, M. 'Social media bot detection with deep learning methods: a systematic review' *Neural Computing and Applications,* Vol 35 No. 12 pp.8903-8918.

[12] Aljabri, M., Zagrouba, R., Shaahid, A., Alnasser, F., Saleh, A., and Alomari, D. M. 'Machine learning-based social media bot detection: a comprehensive literature review' *Social Network Analysis and Mining,* Vol 13 No.20

[13] Ellaky, Z., Benabbou, F., and Ouahabi, S. 'Systematic literature review of social media bots detection systems' *Journal of King Saud University-Computer and Information Sciences*, Vol 35 No. 5

[14] Iryna, B., and Marta, O. 'Review On Social Media Bot Detection using Machine Learning' in 2025: Proceedings of the International scientific and practical conference "Science and society: tools of modern innovative development" *Bilbao, Spain. International Science Group*

[15] TOMA, R. Graph Neural Networks for Tracing Coordinated Bot Activity

[16] Guyan, Q., Liu, Y., Liu, J., and Zhang, P. 'PEGNN: Peripheral-Enhanced graph neural network for social bot detection' *Expert Systems with Applications* Vol 278 No. C DOI: https://doi.org/10.1016/j.eswa.2025.12729

[17] Arranz-Escudero, O., Quijano-Sanchez, L., and Liberatore, F. 'Enhancing misinformation countermeasures: a multimodal approach to twitter bot detection' *Social Network Analysis and Mining* Vol 15 No. 1 DOI:10.1007/s13278-025-01435-w

[18] Javed, D., Jhanjhi, N. Z., Khan, N. A., Ray, S. K., Al-Dhaqm, A., and Kebande, V. R. 'Identification of Spambots and Fake Followers on Social Network via

Interpretable AI-based Machine Learning' *IEEE Access*

[19] Lopez-Joya, S., Diaz-Garcia, J. A., Ruiz, M. D., and Martin-Bautista, M. J. 'Dissecting a social bot powered by generative AI: anatomy, new trends and challenges' *Social Network Analysis and Mining*, Vol 15 No. 7

[20] Ghosh, D., Boettcher, W., Johnston, R., and Lahiri, S. 'Bot Identification in Social Media' Available Online at: https:// arXiv preprint arXiv: 2503.23629. (Accessed 15 March 2025)

[21] Baig, D. 'Bot detection using a machine learning adaptive transfer approach' *Emerging Learning Technologies* Vol 1 No.1 pp.20-29.

[22] TALHA, Z. Enhancing Social Network Security: Machine Learning-Based Bot Detection [online] University of 8 Mai 45 Guelma http://dspace.univ-guelma.dz/jspui/handle/123456789/16472

[23] Lopez-Joya, S., Diaz-Garcia, J. A., Ruiz, M. D., and Martin-Bautista, M. J. 'Exploring social bots: A feature-based approach to improve bot detection in social networks' Available Online at: https:// arXiv preprint arXiv: 2411.06626. (Accessed 14 April 2025)

[24] Wu, Q., Yang, Y., He, B., Liu, H., Yang, R., and Liao, Y. 'BotSCL: Heterophily-Aware Social Bot Detection with Supervised Contrastive Learning' in 2025: *Proceedings of the International Conference on Pattern Recognition Springer*, Cham pp.53-68