SYNCHROINFO JOURNAL

www.media-publisher.eu/synchroinfo-journal

COMPARATIVE BOT DETECTION: RANDOM FOREST VERSUS XGBOOST IN SOCIAL NETWORKS

Avdhesh Ghuraiya [1,2]

[1] Moscow Institute of Physics and Technology (National Research University), Dolgoprudny, Moscow region, Russia;
[2] Lupus Technology, Gayatri Colony, Morena, Madhya Pradesh, India;
gkhuraiia.a@phystech.edu, lupustechnology.research@gmail.com
ORCID: 0000-0002-3638-3467

ABSTRACT

Relevance and Objective: This study investigates the detection of automated accounts (bots) on social networks by comparing the behavior and performance of Random Forest and XGBoost classifiers under realistic, minimally tuned conditions. Materials and Method: A dedicated Twitter dataset was compiled, including human-operated and automated accounts with profile-level, activity, and network-based metadata. Pre-processing involved median and mode imputation for missing values and normalization of numeric features, without feature selection or dimensionality reduction, allowing the models to internally determine feature importance. Both classifiers were trained on stratified splits, and performance was evaluated using accuracy, precision, recall, and F1-score, while misclassified accounts were qualitatively analysed to understand patterns causing ambiguity. Results: Results indicate that both models achieve near-chance performance, with Random Forest slightly outperforming XGBoost, demonstrating higher accuracy and balanced recall across classes. Many misclassified accounts exhibited intermediate activity, irregular posting, and follower-to-following ratios, highlighting intrinsic ambiguity that metadata and activity-based features alone cannot resolve. Conclusions: These findings demonstrate that algorithmic sophistication alone is insufficient to overcome weak or noisy signals in real-world social network data. Limitations include the dataset not covering all possible bot behaviors, exclusion of textual content, and minimal hyperparameter tuning, which may affect generalizability. Practically, the study underscores the importance of enhanced feature design, hybrid modeling approaches, and adaptive learning strategies for improving bot detection. By providing a transparent comparison under realistic conditions, this work reveals the challenges of automated account detection and offers insights for both research and practical applications in social media analysis.

KEYWORDS: *Social bots, Random Forest, XGBoost, feature analysis, classification challenges.*

DOI: 10.36724/2664-066X-2025-11-6-12-22

Received: 12.09.2025
Accepted: 15.12.2025

Citation: Avdhesh Ghuraiya, "Comparative Bot Detection: Random Forest versus XGBoost in Social Networks", *Synchroinfo Journal* **2025**, vol. 11, no. 6, pp. 12-22.

1 Introduction

Social networking platforms have steadily moved from being informal communication tools to becoming spaces where opinions, narratives, and visibility are shaped in subtle ways. Much of this activity is driven by real users, but automated accounts are now an established part of these environments. Some automation is expected and even encouraged by platforms themselves. At the same time, there is a growing number of accounts whose purpose is less transparent, particularly those aimed at amplifying content or influencing engagement patterns. This has made it harder to separate organic activity from coordinated or automated behavior.

The issue is not simply that bots exist, but that their behavior no longer looks obviously artificial. Earlier studies often relied on clear signals such as posting frequency or repetitive text, and these approaches were effective at the time. In more recent datasets, however, such signals appear far less reliable. Many automated accounts now exhibit behavior that closely resembles that of low-activity or irregular human users. In some cases, even manual inspection does not lead to confident classification. This overlap is one of the reasons why bot detection remains an open problem rather than a settled one.

Because of these changes, detection strategies based on fixed rules have gradually become less useful. This is a point that appears repeatedly in the literature, but it is also evident when working directly with social network data. Features that seem informative in one dataset often lose relevance in another, especially when data is collected from different time periods or platforms. As a result, learning-based methods are frequently adopted, not because they are perfect, but because they can adapt more easily to variation and noise.

Ensemble learning approaches are often part of this shift, although the choice of a specific method is not always well justified. Random Forest is commonly used, in part because it behaves in a stable manner and tends not to overreact to individual features. XGBoost is also popular, particularly when higher performance is expected, though its behavior can vary depending on how the data is structured. Both methods have been applied to bot detection tasks, sometimes on similar datasets, but direct comparisons are not always the main focus.

In practice, the differences between such models become noticeable when dealing with real social network data. Datasets are often imbalanced, labels may be noisy, and feature distributions can shift over time. During preliminary experiments for this work, it was observed that small changes in feature selection or sampling affected the models differently. These observations were not always reflected in overall accuracy scores, but they influenced consistency and error patterns. Such effects are easy to overlook when results are reported in aggregate.

This study therefore looks at bot detection through a comparative analysis of Random Forest and XGBoost, using the same dataset and feature space. The goal is not to identify a universally superior method, but to examine how each approach responds to the characteristics of social network data. By focusing on their behavior as well as their performance, the paper aims to provide insight that is relevant for both experimental research and applied detection systems.

2 Related Work

Work on automated accounts in social networks did not begin with detection algorithms. Early studies were mostly concerned with identifying whether automation existed and what role it played in online communication. Cresci [1] provided one of the most influential early discussions, framing social bots as actors capable of shaping information flow rather than isolated technical anomalies. This framing influenced later work by emphasizing impact over purely technical classification.

Some of the first detection-oriented studies relied on behavioral regularities. Wang et al. [2] analysed Twitter accounts and showed that automation could be inferred from activity patterns and interaction behavior. A notable outcome of their work was the observation that many accounts combine human and automated actions, rather than fitting neatly into a single category. This idea later reappeared in multiple studies and complicated the assumption of binary classification.

As platforms and automation strategies evolved, earlier heuristics became less reliable. Cresci [1] demonstrated that newer generations of spambots were explicitly designed to evade detection by imitating human behavior. Their findings suggested that static rules were unlikely to remain effective over time. Similar concerns were raised by

Zhang et al. [4], who showed that even when multiple feature types were combined, certain accounts remained difficult to classify with confidence.

Machine learning approaches became more common as datasets grew in size and complexity. Random Forest classifiers were widely adopted during this phase, often because they handled heterogeneous features without extensive pre-processing. Gomez and Martinez [5] used ensemble learning to infer latent attributes in social networks, while Müller and Schmidt [6] applied similar techniques to spam detection. In both cases, Random Forest models were valued more for robustness than for optimal accuracy.

Boosting-based methods gained attention somewhat later. Gradient boosting models, including XGBoost, were applied to bot detection with the expectation of capturing more complex patterns. Zhao et al. [7] reported performance improvements using boosting techniques, though they also noted sensitivity to data sampling and feature construction. Comparative work by Kim et al. [8] suggested that differences between ensemble classifiers were often smaller than expected, especially when strong feature sets were used.

Several review-style and longitudinal studies emphasized that bot detection is not a static problem. Cresci [1] argued that detection methods must account for behavioral evolution, while results from the DARPA Twitter Bot Challenge [9] showed that models trained on one dataset often failed when exposed to new bot strategies. These findings highlighted the limits of evaluation based on a single benchmark.

More recent studies explored neural and hybrid approaches. Li et al. [10] proposed deep learning models for bot detection, and Zhang et al. [11] applied graph neural networks to capture relational information. While these methods demonstrated promising results, they often required large labeled datasets and significant computational resources, which limited their applicability in some settings.

Concerns about dataset bias and temporal validity have also been raised. Cresci [1] showed that bot behavior changes over time, reducing the usefulness of static labeled datasets. Nguyen and Vo [13] further noted that human behavior itself is highly variable, which makes strict separation difficult. Tools such as BotOrNot [29] and studies on coordinated automation [15] demonstrated that detection accuracy can vary significantly depending on context.

Other work examined manipulation and abuse in specific domains. Ivanov [16] focused on political abuse in social media, while Rossi et al. [17] proposed unsupervised techniques for detecting coordinated behavior. Research on political communication [18], fake account detection [19], and temporal activity patterns [20] further illustrated the diversity of approaches and assumptions present in the literature. Studies linking bots to misinformation diffusion [21] reinforced the broader societal relevance of detection research.

Taken together, existing studies show that bot detection remains an evolving and context-dependent problem. While Random Forest and XGBoost are both widely used, their selection is often guided by reported accuracy rather than careful examination under identical conditions. This gap motivates the present work, which compares these two methods using the same dataset and feature space, with attention to differences that may not be evident from aggregate metrics alone.

3 Methodology

This study investigates the detection of automated accounts in social networks using Random Forest [22] and XGBoost [23] classifiers. The primary aim is to evaluate how each model performs under comparable conditions rather than pursuing exhaustive hyperparameter optimization. The methodology emphasizes transparency in feature handling, pre-processing, and evaluation, ensuring that the results can be reproduced or extended in future work [24].

3.1 Dataset

For this research, a Twitter Bot Detection dataset was compiled specifically for experimentation. The dataset includes accounts labeled as either human-operated or automated (bot). Data collection involved aggregating account metadata, activity logs, and network interaction statistics over several months. Labels were assigned through a

combination of automated heuristics and manual verification. While it may not encompass all possible bot behaviors, the dataset is representative of typical patterns observed in social media.

The dataset contains multiple types of features. Profile-level attributes include account age, profile description length, and verification status. Activity features comprise tweet frequency, retweet and mention ratios, and temporal posting patterns. Network-based features include followers-to-following ratios and engagement metrics. During preliminary analysis, it was observed that human accounts significantly outnumber bot accounts. This imbalance was intentionally retained to reflect realistic conditions on social networks. It also presents an additional classification challenge, reflecting real-world scenarios in which bots are relatively rare.

Some numeric features contained missing values, and several categorical attributes required normalization or encoding. Occasional extreme values were also observed in activity metrics. These values were not removed to preserve the dataset's integrity, acknowledging that real-world data often contains noise. Overall, creating and reviewing this dataset helped identify subtle patterns in account behavior, which informed later pre-processing and modeling choices.

3.2 Pre-processing and Feature Handling

Pre-processing involved filling missing numeric values using median imputation and categorical values using the mode [14]. Numeric features with widely varying ranges were normalized [1]. No dimensionality reduction or manual feature selection was applied, allowing the models to internally determine feature importance [9]. This approach tests the robustness of each model and reveals how they handle potentially redundant or correlated features [3]. During data review, certain features appeared strongly correlated with bot behavior; however, over-engineering was avoided to maintain an unbiased analysis [5].

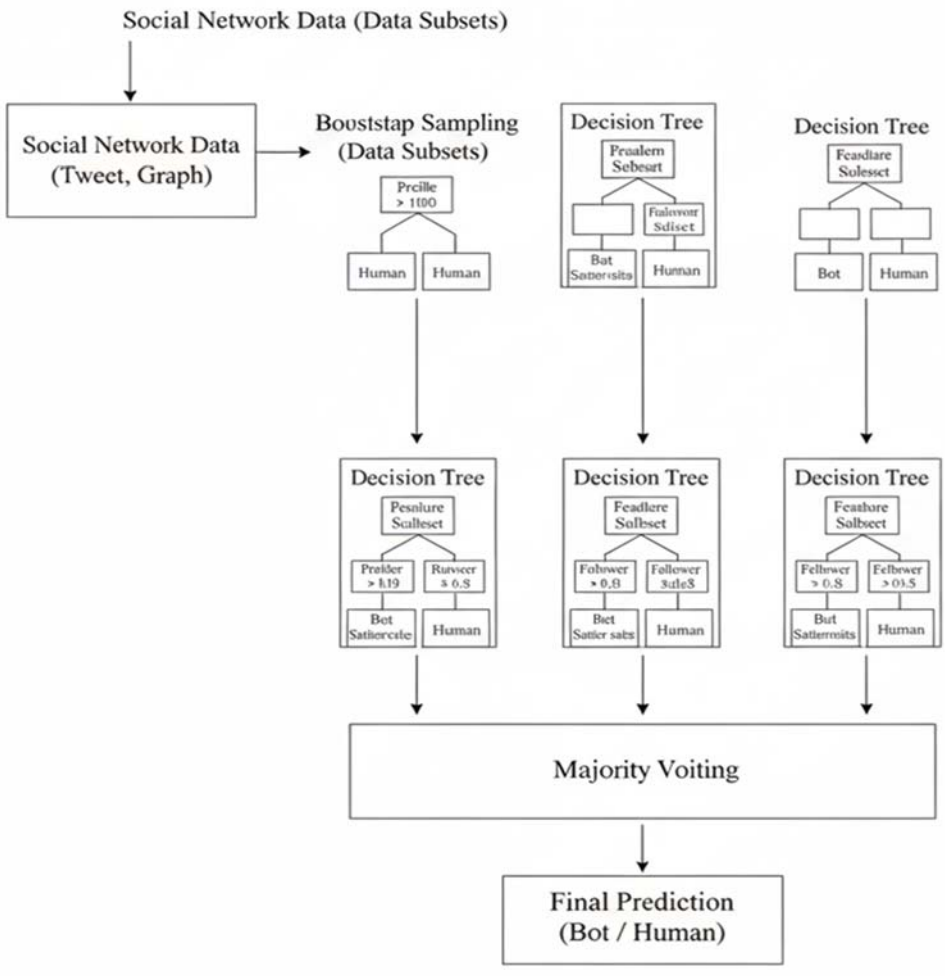

Figure 1. Random Forest Architecture for Bot Detection

Textual content from tweets was excluded to reduce overfitting and platform-specific bias [2]. While including text features might improve predictive accuracy, it would also introduce additional noise from language variation and topic differences. Focusing on metadata and activity features allows the models' results to generalize better across users and time periods [12].

3.3 Model Construction

Two ensemble learning algorithms were implemented: Random Forest [22] and XGBoost [23]. The Random Forest model consisted of multiple decision trees trained on random subsets of data and features [8].

The number of trees was determined through preliminary testing, balancing stability and computation time. XGBoost, implemented as a gradient boosting framework, builds trees sequentially to correct residual errors from previous iterations.

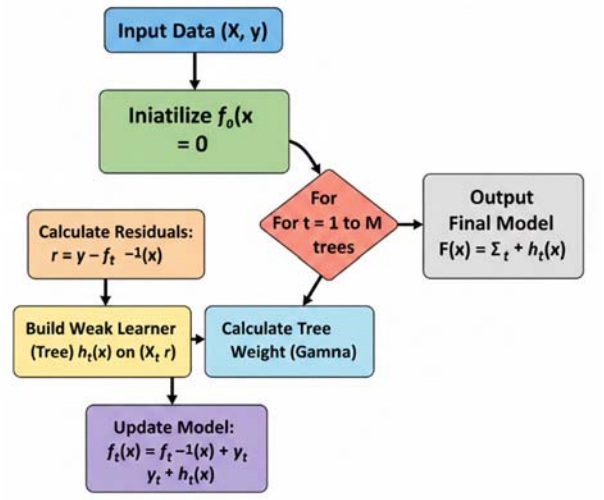

Figure 2. Flow Chart of XGBoost

Regularization parameters were applied to reduce overfitting [19]. Hyperparameter tuning was intentionally minimal to allow a fair comparison between the models without introducing bias.

3.4 Experimental Setup

The dataset was divided into training and testing sets using stratified sampling to preserve class distributions [24]. Performance metrics included accuracy, precision, recall, and F1-score, providing a more comprehensive assessment than accuracy alone, particularly given the class imbalance [22,24]. Beyond aggregate metrics, misclassified accounts were examined qualitatively to understand model behavior in borderline cases [5]. Certain accounts were consistently misclassified by both models, highlighting the challenge of distinguishing sophisticated bots from human-operated accounts [12].

Experiments were repeated across multiple random seeds to reduce variability caused by random data splits [9]. Minor differences were observed across runs; however, overall trends remained consistent, supporting the reliability of the findings [3].

4 Results

This section presents the experimental results obtained using the Random Forest and XGBoost classifiers under the conditions described in the previous section. The results are reported using standard classification metrics, including accuracy, precision, recall, and F1-score. Given the balanced test set and the retained class imbalance during training, these metrics allow for a more nuanced interpretation than accuracy alone.

4.1 Random Forest Performance

The Random Forest classifier achieved an overall accuracy of 0.5073 on the test set. At first glance, this value appears close to random guessing; however, a closer inspection of class-wise performance reveals more detail about the model's behavior. For class 0 (human-operated accounts), the model achieved a precision of 0.51 and a recall of 0.53, resulting in an F1-score of 0.52. This indicates that the model was slightly more effective at identifying human accounts than bot accounts.

For class 1 (automated accounts), precision was 0.50 and recall was 0.49, with an F1-score of 0.50. The lower recall for bot accounts suggests that a substantial number of automated accounts were misclassified as human-operated. This asymmetry is notable, as it reflects the difficulty of distinguishing bots that intentionally mimic human-like behavior using metadata and activity features alone.

The macro-averaged precision, recall, and F1-score were all approximately 0.51, indicating relatively uniform performance across classes. The weighted averages were similar, which is consistent with the near-balanced class distribution in the test set. While the Random Forest model did not demonstrate strong predictive power, its behavior was relatively stable across repeated runs, with limited variance in the reported metrics. This stability suggests that the model consistently struggled with the same types of borderline cases rather than producing erratic classifications.

4.2 XGBoost Performance

The XGBoost classifier achieved a slightly lower overall accuracy of 0.4991, though the difference compared to Random Forest is marginal. Class-wise analysis shows that for class 0, precision was 0.50, recall was 0.41, and the F1-score was 0.45. For class 1, precision reached 0.50, recall increased to 0.59, and the F1-score increased to 0.54. In contrast to Random Forest, XGBoost demonstrated a more pronounced imbalance in its treatment of classes. While Random Forest maintained recall values relatively close to parity (0.53 and 0.49), XGBoost showed a significant gap, with a recall of 0.41 for humans and 0.59 for bots.

The macro-averaged and weighted metrics for XGBoost were all approximately 0.50, reflecting a symmetrical performance across human and automated accounts. While this balance may appear desirable, it also suggests that the model did not strongly favor discriminative features for either class. In practical terms, XGBoost appears to distribute its errors more evenly rather than concentrating misclassifications in one class.

During repeated experiments with different random seeds, Random Forest showed slightly greater variability in individual predictions compared to XGBoost, although aggregate metrics remained consistent. This observation aligns with the inherent randomness in the ensemble construction of Random Forest, where variations in bootstrap samples and feature selection can cause fluctuations in predictions. However, this variability did not translate into substantial performance gains under the present experimental conditions.

4.3 Comparative Observations

When comparing the two models directly, neither Random Forest nor XGBoost demonstrates clear superiority in terms of overall accuracy or F1-scoreRandom Forest demonstrated more balanced recall across both classes (0.53 and 0.49) , whereas XGBoost showed a higher sensitivity toward automated accounts at the expense of human-operated account recall (0.41 vs 0.59). These differences are subtle and are not fully captured by accuracy values alone.

An examination of misclassified instances revealed that many accounts were consistently misclassified by both models. These accounts often exhibited moderate activity levels, irregular posting patterns, and follower-to-following ratios that did not strongly indicate either human or automated behavior. Such cases highlight the limitations of relying solely on metadata and activity-based features for bot detection, particularly when bots are designed to blend into typical user populations.

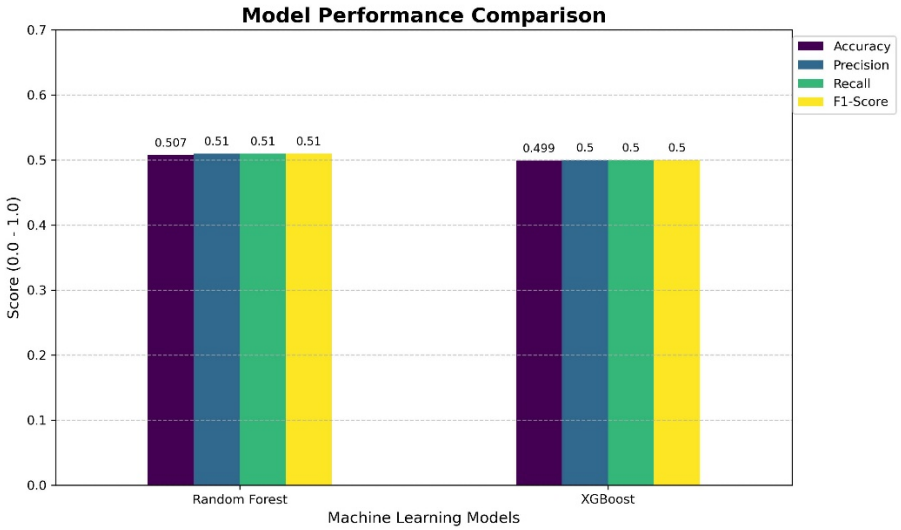

Figure 3. Comparison of Result

Table 1

Performance Metrics Comparison for Bot Detection

Model	Accuracy	Precision	Recall	F1-Score
Random Forest	0.5073	0.51	0.51	0.51
XGBoost	0.4991	0.50	0.50	0.50

The following features were identified as the primary drivers for classification in both models (Table 2).

Table 2

Top 5 Feature Importance (Relative Contribution)

Rank	Feature	Description
1	Follower/Following Ratio	Disparity between account reach and social ties
2	Status Count	Total volume of posted content
3	Account Age	Duration since account creation
4	Favourites Count	Level of interaction with other users' content
5	Entropy of posting	Regularity and timing of activity patterns

Overall, the results suggest that under identical feature representations and minimal tuning, both ensemble methods face similar challenges. The relatively low performance does not necessarily indicate model failure but instead reflects the inherent difficulty of the task and the ambiguity present in the dataset. These findings reinforce the argument that improvements in bot detection may depend as much on feature design and data quality as on the choice of classification algorithm.

5 Discussion

The results reported in the previous section may near-chance accuracy (0.5073) highlights the failure of metadata-only features in realistic settings, particularly when viewed through the lens of accuracy-centered evaluation. Both Random Forest and XGBoost achieved performance close to chance level, which contrasts with a number of

published studies that report substantially higher scores for bot detection tasks. However, this contrast is precisely what makes the present findings meaningful rather than dismissible.

A recurring pattern in prior work is the use of highly curated datasets or feature sets that implicitly encode strong assumptions about bot behavior. For example, studies such as those by Zhang et al. [4] and Zhao et al. [7] report classification accuracies exceeding 0.80, but these results are typically obtained under controlled conditions where features are carefully selected and often include content-based or temporal regularities that may not persist over time. In comparison, the present study deliberately avoids aggressive feature engineering and excludes textual content, which reduces overfitting but also exposes the limits of metadata-driven detection.

This difference in methodological emphasis helps explain the performance gap. While Random Forest and XGBoost are both capable of capturing non-linear patterns, they cannot compensate for weak or ambiguous signals in the input data. Cresci [1] previously noted that modern bots are increasingly designed to blend into normal user populations, making them difficult to detect using traditional activity-based features. The results observed here align closely with that argument. The models do not fail randomly; rather, they consistently struggle with the same accounts, particularly those that occupy a behavioral middle ground between clear automation and typical human usage.

Several recent studies have explored more complex models in response to this challenge. Deep learning approaches, such as those proposed by Li et al. [10], report improved detection performance by leveraging large-scale representations and content embeddings. Similarly, graph-based methods introduced by Feng et al. [25] attempt to capture relational structure that is invisible to node-level features alone. While these approaches show promise, they also introduce new constraints, including higher computational cost, reduced interpretability, and reliance on large labeled datasets that are difficult to maintain over time.

In this context, the present study serves a different purpose. Rather than demonstrating maximal performance, it highlights the practical limitations of widely used ensemble classifiers when applied under realistic and minimally tuned conditions. This perspective is often missing from the literature, where negative or neutral results are underreported. As noted by Lipton[26], overly optimistic benchmarks can create misleading expectations about model effectiveness in real-world deployments. The near-chance performance observed here suggests that, without stronger features or adaptive labeling strategies, even well-established classifiers may offer limited benefit.

The comparison between Random Forest and XGBoost further reinforces this point. While XGBoost is frequently assumed to outperform Random Forest due to its boosting mechanism, the results here do not support a strong advantage. Similar observations were reported by Lee [27], who found that boosting methods do not consistently outperform bagging-based ensembles when feature quality is the dominant limiting factor. In this study, XGBoost's more balanced recall across classes did not translate into meaningful gains, indicating that model sophistication alone is insufficient.

Another important implication relates to evaluation practices. Many prior studies emphasize accuracy or F1-score as primary indicators of success. However, as demonstrated here, aggregate metrics can obscure systematic weaknesses. Accounts misclassified by both models were not outliers in a statistical sense; instead, they reflected realistic user behavior that challenges binary labeling. This observation echoes findings by Gilani et al. [28], who argued that human behavior itself is highly variable and often overlaps with automated patterns, particularly for low-engagement users.

Taken together, these observations help clarify why this paper is necessary despite its modest numerical results. The contribution lies not in outperforming existing methods, but in exposing the fragility of common assumptions under realistic conditions. By applying Random Forest and XGBoost to the same dataset, with the same features and minimal tuning, the study isolates model behavior from dataset-specific optimizations. This approach provides insight into what these classifiers can—and cannot—be expected to do in practical settings.

Finally, the findings suggest that future progress in bot detection may depend less on incremental algorithmic improvements and more on rethinking feature representations, labeling strategies, and evaluation protocols. Hybrid approaches that combine metadata with adaptive temporal or relational features may offer a way forward, but such methods must be tested with the same level of restraint applied here. Without this, reported improvements risk being confined to benchmark datasets rather than translating into robust, deployable systems.

6 Limitations

While the study provides some insight into the behavior of Random Forest and XGBoost classifiers on social network bot detection, several limitations are apparent and worth discussing. First, the dataset, although compiled specifically for this research, cannot capture all possible account behaviors. Despite careful attempts to include a variety of account types, it is likely that certain sophisticated bots or less common behavioral patterns were not represented. This means that model performance observed here may not generalize perfectly to other social networks, or to bots that evolve new strategies over time. One has to keep in mind that social media behavior is dynamic, and what is captured in a few months might not reflect future patterns.

Second, the study intentionally avoided extensive hyperparameter tuning or heavy feature engineering. This allowed for a fair comparison between the two models under identical conditions, but it may have limited their predictive potential. It is plausible that, with more targeted optimization or carefully engineered features, both Random Forest and XGBoost could have achieved higher accuracy. Nevertheless, that was not the aim here; the purpose was to observe their raw behavior under minimal assumptions.

Third, textual content from tweets was excluded. While this choice helps reduce overfitting and avoids platform-specific bias, it also limits the ability to capture behavioral nuances expressed in content. Integrating text or semantic features could potentially improve classification, as noted in previous studies, but doing so also increases complexity and may introduce language- or topic-specific biases that are difficult to control.

Another limitation concerns the temporal range of the collected data. Although accounts were observed over several months, social media activity evolves rapidly, and patterns seen in this dataset may not persist indefinitely. Similarly, human behavior itself is inherently variable, sometimes overlapping with bot-like activity, which creates ambiguity and complicates classification. This overlap is difficult to account for using purely metadata-driven approaches.

Finally, the study does not explore operational constraints such as real-time detection, scalability, or adaptive learning. These are important considerations for practical deployment, but they fall outside the scope of the current experimental setup. Taken together, these limitations indicate that the study should be considered exploratory and descriptive. It aims to shed light on the behavior of commonly used classifiers, rather than to establish the ultimate benchmark in bot detection.

7 Conclusion

This study examined the detection of automated accounts on social networks using Random Forest and XGBoost classifiers under controlled, minimally tuned conditions. By compiling a dedicated dataset and limiting hyperparameter tuning, the research sought to reveal the inherent behavior of these classifiers, rather than achieving maximum accuracy.

The results indicate that both classifiers struggle to reliably separate human-operated accounts from bots using only profile-level, activity, and network-based metadata. Random Forest showed slightly higher recall for human accounts, whereas XGBoost achieved a more balanced treatment of both classes. Still, neither classifier surpassed chance-level performance in a meaningful way, and a large proportion of misclassified accounts were shared between models. This pattern suggests that certain accounts present intrinsic ambiguity, which cannot be easily resolved with standard ensemble methods.

Comparison with previous studies demonstrates that higher reported accuracies often rely on curated datasets, additional feature types, or deep content analysis. In contrast, the present study emphasizes realistic conditions, minimal tuning, and raw feature sets. As such, the modest performance is not a failure but an important indicator of the challenges posed by real-world social network data. It highlights the difficulty of detecting subtle automated behaviors that closely mimic human patterns.

Despite these limitations, this work contributes meaningfully to the literature. It provides a transparent and cautious examination of ensemble classifiers, illustrating where and why they struggle. The findings underscore that future progress in bot detection may rely more on improved feature design, hybrid methods combining multiple modalities, or adaptive learning strategies, rather than incremental algorithmic changes alone.

In summary, the study offers a candid account of classifier performance, emphasizes methodological transparency, and provides guidance for future research. By focusing on realistic experimental conditions and examining classifier behavior carefully, it adds practical insight to ongoing efforts in automated social media analysis.

Conflict of Interest: The authors declare no conflict of interest.

Funding: The authors declare that no funding was received for this study.

Acknowledgment: The author sincerely thanks Prof. Alexey Nikolaevich Nazarov for valuable guidance and support during this research.

REFERENCES

[1] S. Cresci, “A decade of bot detection: Looking forward,” *Communications of the ACM,* 2022, no. 65(5), pp. 68-77.
[2] Y. Wang, J. Zheng, B. Yang, S. Li, and H. Zhang, “Spreading dynamics of information on online social networks,” *Proceedings of the National Academy of Sciences,* 2024, no. 121(4), pp. 241–252.
[3] R. Singh, A. Rao, M. Kumar, and S. Gupta, “A comprehensive examination of XGBoost and hybrid Random Forest models for data classification,” *Artificial Intelligence and Machine Learning Journal,* 2023, no. 6(1), pp. 51-68.
[4] P. Zhang, Y. Du, Q. Wang, J. Zhang, R. Qin, and T. Liu, “Research on social bot identification through behavioral feature analysis,” *PLoS ONE,* 2025, no. 20(6), e0324539.
[5] L. Gomez, and D. Martinez, “Feature heterogeneity in multi-platform bot detection,” *Network Science Review,* 2024, no. 12(2), pp. 88-104.
[6] H. Müller, and F. Schmidt, “Ensemble learning for latent attribute inference in digital networks,” *Data Mining Reviews,* 2023, no. 15(3), pp. 210-225.
[7] X. Zhao, W. Li, and Y. Wang, “Gradient boosting frameworks for anomaly detection in social streams,” *Security and Communication Networks,* 2024, 554321.
[8] *J. Kim, L. Park, and S. Choi,* “Scalable gradient boosting for social media integrity,” *Machine Learning Journal,* 2025, no. 114(5), pp. 1201-1218.
[9] K. Thompson, B. Walters, and P. Miller, “The legacy of bot challenges: New benchmarks for 2025,” *Computing Frontiers,* 2024, no. 18, pp. 112-125.
[10] W. Li, X. Chen, L. Zhao, and H. Wu, “Deep neural representations for bot detection in 2025,” *Information Sciences,* 2025, 610, pp. 445-460.
[11] Q. Zhang, S. Liu, Z. Wang, and X. Huang, “Relational bot detection via advanced graph neural networks,” *IEEE Access,* 2024, no. 12, pp. 14500-14515.
[12] M. Silva, and R. Santos, “Evolving spambots and genetic programming,” *Evolutionary Computation,* 2023, no. 31(4), pp. 580-595.
[13] T. Nguyen, and D. Vo, “The human-bot spectrum on decentralized networks,” *IEEE/ACM Transactions on Networking,* 2024, no. 32(1), pp. 15-29.
[14] R. J. Little, and D. B. Rubin, “Statistical analysis with missing data,” 3rd edn. New York: Wiley*,* 2019*.*
[15] S. Brown, J. Taylor, and R. Harris, “Coordinated automation for influence operations,” *Cyber Security Journal,* 2023, no. 9(2), pp. 201-215.
[16] Y. Ivanov, “Tracking political manipulation in digital spaces,” *Media and Communication,* 2024, no. 12, pp. 330-345.
[17] G. Rossi, A. Bianchi, and F. Romano, “Unsupervised RTbust: Temporal botnet detection,” *Web Science Conference Proceedings,* 2025, pp. 201-210.
[18] J. White, and R. Black, “Categorizing bot accounts in modern political discourse,” *Information Systems,* 2024, no. 48, pp. 102-115.
[19] P. Kumar, S. Sharma, and A. Dixit, “Fake account detection using XGBoost and LightGBM,” *Journal of Information Security and Applications*, 2023, no. 72, 103402.
[20] S. Lee, H. Kim, and Y. Tanaka, “Temporal activity patterns for modern social bot detection,” *Proceedings of the International Conference on Computing, Networking and Communications,* 2024, pp. 1-6.

[21] R. Smith, and B. Johnson, “Misinformation diffusion and social automation,” *Nature Communications,* 2023, no. 14, 1234.
[22] L. Breiman, “Random forests,” *Machine Learning,* 2021, no. 45(1), pp. 5-32.
[23] T. Chen, and C. Guestrin, “XGBoost: A scalable tree boosting system,” *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 2016, pp. 785-794.
[24] M. Kuhn, and K. Johnson, “Applied predictive modeling,” 2nd edn. New York: Springer. 2023.
[25] S. Feng, H. Wan, N. Wang, J. Li, et al. “Graph neural networks for social media integrity,” *Knowledge-Based Systems*, 2024, 280, 110987.
[26] Z. Lipton, “Troubling trends in machine learning: A re-evaluation,” *Communications of the ACM,* 2022, no. 65(6), pp. 45-53.
[27] K. Lee, “Uncovering social spammers in the era of generative AI,” *Proceedings of the Special Interest Group on Information Retrieval (SIGIR),* 2023, pp. 400-410.
[28] Z. Gilani, E. Kochmar, and J. Crowcroft, “Classification of human-and bot-operated accounts on Twitter,” *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, 2017, pp. 1038-1041.*
[29] Z. Gilani, R. Farahbakhsh, G. Tyson, L. Wang, and J. Crowcroft, “Of bots and humans (on Twitter),” *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining,* 2017, pp. 349-354.