Smart Detection: A System for Identifying BOT Activities on Social Media

Manjeet Singh
1.a. Research scholar Amity Business School, Amity University Chhattisgarh
1.b. Manager software engineering, Clear Trail Technologies Pvt Ltd. SDF No. K-12, NSEZ, Noida, UP 201305
https://orcid.org/0009-0000-2157-1388
Annapurna Metta
Assistant Professor, Amity Business School, Amity University Chhattisgarh
https://orcid.org/0009-0009-2015-6815
Satyendra Patnaik
Professor & Dean, JSS University Noida: Noida, Uttar Pradesh, IN

Abstract

The increasing number of automated bot accounts on social networking platforms such as Twitter creates a daunting challenge as these bots help in the spread of fake news, influence public perception and shift the online narrative. Banishment of frauds is crucial within the realm of social networks and involves the recognition and differentiation between human and bot accounts. The research seeks to propose an advanced machine learning framework for bot detection on Twitter that integrates heuristics approaches, sentiment analysis and ensemble learning approaches. The procedure includes the extraction of user profiling, content and metadata of the tweet such as sentiment polarity and subjectivity required for the determination of the unique features that separate bots from humans. The class imbalance issue was rectified during data normalization using SMOTE (Synthetic Minority Over-sampling Technique) so as to create a more just environment for training. The ensemble model consisted of Random Forest, XGBoost, Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Gradient Boosting which were enrolled as classifier in the process so as to increase the quality of the classification process. The model accuracy, precision, recall, F1-score and confusion matrix evaluation were all included in assessing the models performance. The approach we used was able to classify whether an account was a bot or user with accuracy of 95.36%,03 precision of 93.47%, recall of 94.65%, and F1 of 94.06, thus proving effectiveness of the method used for the classification. To address the arising issue of bot attacks, it can be seen that the combination of sentiment analysis and supplementing it with advanced ensemble learning techniques can significantly improve the detection and mitigation of such social media attacks.

Keywords: SMOTE, Ensemble Learning, Logistic Regression, Bot Mitigation, Sentiment Analysis

1. Introduction

The advent of social networking sites like Twitter has revolutionized the way people converse and share information, as it allows people to communicate with a global audience at the same time without any barriers. This increased interaction has led to the increasing prevalence of bot accounts, which pose a major threat in the form of information fraud, sentiment manipulation, and artificially increasing trends. Studies from recent times show that bots played an important role in spreading fake news during the american elections of 2016 and also during the COVID 19 pandemics thus diminishing the trustworthiness of social networks [1, 2]. At least 15 percent of the active accounts on Twitter are auto bot accounts, marking a, sizable percentage of Twitters user accounts [3]. The complexity lies in the accurate identification of real users from auto bot accounts as they are becoming better at mimicking humans. Especially in the context of rapidly changing bot behavior and the large volume of data generated on social media sites, it is accepted that traditional detection methods using manual heuristics or basic machine learning models tend often to fall short in many respects [4]. As a solution to these problems, advanced techniques such as complex machine learning models fused with natural language processing NLP approaches have emerged as effective tools in bot detection. Sentiment analysis is confirmed to be very useful in establishing whether there is a large and significant difference between human content and bot content and therefore helps in refining the classification further [5-7]. At the same time, ensemble learning which is the use of multiple models to achieve higher prediction accuracy has been gaining popularity as robust and effective solutions to more complex classification problems [8, 9]. This paper introduces a concrete approach with the help of sentiment analysis and ensemble learning for a robust and accurate Twitter bot detection system. The method aims to extract relevant features from Twitter including user profile, tweet information and even metadata such as sentiment polarity and subjectivity to classify Twitter users as human or bot. The model also aimed to solve the problem of class imbalance by using SMOTE which increases classification ability of the model on unseen data. Model accuracy reported ranges above 95%, where the model uses Random Forest, XGBoost, KNN, SVM and Logistic Regression as base classifiers and Gradient Boosting as meta learner. Considering the metrics of the model including precision, recall and F1-score, the values achieved provide a comprehensive outlook of the model performance.

The main contributions of the paper are as follows:

- Twitter bot detection system is presented that combines sentiment analysis, heuristic criteria, and sophisticated ensemble learning algorithms, offering a strong solution for recognizing artificial accounts.

- The problem of class imbalance is handled using SMOTE, guaranteeing that the model is effectively trained to manage heterogeneous datasets with differing bot-human ratios.
- The suggested ensemble model, integrating Random Forest, XGBoost, Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Gradient Boosting Classifier (GBC) attains elevated accuracy, precision, recall, and F1-score, indicating substantial enhancements compared to conventional bot identification techniques.
- Presented a comprehensive assessment of the model's efficacy, emphasizing the proficiency of sentiment analysis and ensemble methodologies in differentiating between real and bot accounts.

Unlike prior research that emphasizes the use of machine learning classifiers for bot detection, our work combines sentiment analysis with ensemble learning, achieving significantly better results.

Though applications such as LSTM and CNNs have been implemented in bot detection, they demand a substantial amount of training data, computational resources, and preprocessing work. In the context of needing real-time bot detection in social media at scale, this work emphasizes resource efficiency without impacting accuracy. The chosen ensemble approach combines various classifiers to use their strengths while minimizing the impact of their individual weaknesses. Random Forest and XGBoost are highly effective at capturing more intricate patterns, while Logistic Regression and SVM provide clearer interpretations. Classification accuracy is improved from deep learning techniques to computationally light ensemble models, reaching 95.36% with ensemble accuracy.

The incorporation of ensemble learning techniques permits the model to accommodate several features at once. Even when bots try to mask the content with NLP methods, the ensemble approach will use multiple classifiers to identify inconsistencies in behavior, content creation, and engagement metrics. Moreover, detecting sentiment analyzes the tone of the content and uncovers inconsistencies, which is a feeble point for bots portraying humans. With metadata information, such as the number of hashtags and retweets, and user activity, the model is capable of recognizing patterns even when content is masked or enhanced through videos and images.

The subsequent sections of this work are organized as follows: Section 2 presents an overview of the relevant research on bot detection and sentiment analysis, emphasizing significant developments and challenges within the discipline. Section 3 delineates the methodology employed in our bot detection strategy, specifying the procedures for data preparation, feature extraction, and model training. Section 4 delineates the experimental configuration, encompassing the hyperparameter optimization procedure and the evaluation metrics utilized to gauge the model's efficacy. Section 5 presents the findings, providing a comparison between the proposed methodology and current techniques to illustrate its efficacy. Section 6 ultimately summarizes the work by encapsulating the principal findings and proposing prospective avenues for future investigation.

This paper presents an innovative and efficient approach to Twitter bot detection by integrating sentiment analysis with sophisticated ensemble learning algorithms, thereby aiding in the preservation of integrity and authenticity within online social networks.

2. Related Work

The issue of flagging automated bot accounts on social media sites has attracted considerable interest in the last decade owing to the ever increasing drifts bots are fetching in online conversations. Various researches have examined a variety of approaches to distinguish between authentic users and bot accounts, including rule-based techniques and machine learning models. Graph-based approaches are becoming more popular because they have the potential to capture network communication patterns with less computational costs compared to flow-based methods [10, 11]. Machine learning algorithms, particularly the random forest model, have been found to be accurate in botnet detection using a DNS query data approach [12] and different elements of social media content [13]. Multilayer frameworks with filtering and classification components have shown some efficacy in the detection of command and control servers [14]. There is an emphasis on fitting models to enable detection of IoT specific botnet [15], and mobile botnet detection systems [16]. Other works have also sought to improve classification accuracy for example lesser feature selection methods recursively [17]. Learning-Based approaches have also proved to be successful in detecting numerous types of botnets on a several platforms.

Various research works have been done on comparing the effectiveness of various algorithms and ensemble selection techniques in botnet detection, with the decision tree and neural network techniques also being highlighted as very accurate [18]. It was established that in addition to bot account features, engineers also worked with human features, such as the ratio of friends to followers, in order to determine whether an account is fake or real, which is the case for both bots and people [19]. Multi layered and multi framework approaches to network intrusion detection systems have been put forth to boost accuracy levels with some of the frameworks, being able to detect P2P botnets with accuracy levels exceeding 98.7% [18]. Two approaches have been successfully developed in the mobile sphere as well. One is behavior analysis and the other is systems with ML integrated in them, wherein the integrated facilities have enabled the detection of botnet applications with an astounding 99.49% accuracy using a logistic regression model [20]. Nevertheless, the changing characteristics of botnets and the continuing conflict of interest between cyber criminals and researchers is still a great deal of progressing case [21].

Table 1 summarizes major studies, techniques, datasets, results, limitations pertaining to the literature review of the automated bot accounts on social media. This systematic overview presents types of algorithms that have been adapted starting from simple heuristics techniques to complex machine learning algorithms for the purpose of detecting bots. Likewise, it shows the performance of a variety of algorithms and techniques for feature selection solving the complex problems pose by botnets in many platforms. It also describes the strengths and the weaknesses of the methods
It highlights the technology development that addresses the as-it-is needs for machine learning algorithms, feature selection approaches, and real-time detection capabilities, as well as the necessary improvements to adapt to the evolving threat landscape.

Table 1: Summary of Key Studies on Bot Detection Techniques, Datasets, Results, and Limitations

Reference/Year	Main Findings	Dataset Used	Results	Limitations
[10]/ (2020)	Two-phase bot detection using unsupervised and supervised ML, robust to zero-day attacks.	Network flow data	High precision in detecting multiple types of bots, including zero-day attacks.	High computational overhead and does not fully capture all communication patterns.
[14]/ (2021)	Behavior-based analysis of botnet traffic with structure/protocolindependent framework.	VPN tunneling technique	F-score up to 92%, falsenegative rate as low as 1.5%.	Limited structure and protocol independence, detection rate not 100%.
[12]/ (2018)	Random forest achieved 90.80% accuracy in detecting botnets using DNS query data.	DNS query data	Random forest outperformed other classifiers.	Limited dataset size, results may not generalize.
[22]/ (2020)	Decision tree, random forest, and CNN performed best in detecting Mirai and Bashlite botnets.	N-BaIoT dataset	CNN outperformed other DL models; high accuracy in binary classification.	Focused on Mirai and Bashlite botnets, limited protocol set.
[16]/ (2015)	Detection of mobile botnets using ML, evaluated for performance and battery impact on Android devices.	Real mobile traffic from Android devices	Effective botnet detection with minimal impact on device performance.	Focused on Android; limited to botnet detection, not other malware.
[23]/ (2023)	Content-based features (POS, sentiment, special characters) used for bot detection on Twitter, with DL outperforming others.	Twitter data	DL outperformed other models when using Information Gain-ranked features.	Did not consider images; focused only on contentbased features.
[24]/ (2021)	High detection rates with reduced training time and complexity; robust to detecting multiple botnet families.	CTU-13, IoT-23 datasets	Higher precision than state-of-theart methods with competitive accuracy.	Focused only on selected graph features, performance on larger datasets unknown.

[25]/ 2018	Compared Naive Bayes, Decision Tree, and Neural Network algorithms and their ensembles for botnet detection.	CTU-13	Decision Tree had better performance; ensemble methods showed enhanced predictions.	Does not mention limitations explicitly in the study.
[26]/ 2018	Applied features used for bot detection to detect fake human identities on social media.	Social media data	Successfully applied bot detection techniques to fake human accounts.	Engineered features for bot detection may not work as effectively for human-created fake identities; limited research on human identity detection.
[18]/ 2019	Proposed a multi-layer detection technique for P2P botnets, with the final layer achieving 98.7% accuracy using Decision Tree classifiers.	Network communication data	Accurate detection of P2P botnets with decision trees and feature selection; reduced network traffic through pre-filtering.	Feature selection process and P2P botnet identification still require improvements.
[20]/ 2016	Used dynamic analysis augmented with machine learning to detect botnet behavior in Android applications.	Drebin Dataset	High detection accuracy for mobile botnet applications using behavioral features and machine learning classifiers.	Limited by 8MB file size in the sandbox environment; sandbox evasion techniques used by some botnets; challenges with comprehensive code coverage in dynamic analysis.
[27]/ 2019	Evaluated seven machine learning algorithms for detecting IoT network attacks, with AdaBoost and ID3 algorithms showing the best performance.	Bot-IoT dataset	AdaBoost had the best overall performance; ID3 had high detection rates for multiple attack types (90%+ for 6 out of 10 attacks).	Does not explicitly mention limitations but suggests future work in evaluating unsupervised algorithms and combining models for improved detection.

[28]/ (2022)	ICA and RF classifier achieved 99.99% accuracy, with a 0.12-second prediction time; ICA crucial for high performance.	N-BaIoT, Aposemat IoT-23.	Outperformed Naive Bayes, SVM, k-NN; 99.99% accuracy.	Further improvements needed for real-time detection and faster prediction.
[29]/ (2023)	RF algorithm outperformed others; real dataset used, focusing on user browsing behavior for fraud detection.	Real user browsing behavior dataset.	RF outperformed others on all evaluation metrics.	Sampled dataset may not be fully representative; computational intensity of some algorithms.
[30]/ (2021)	Scatter search-based DMLP classifier outperformed other models, achieving 100% accuracy and 21.38-second training time.	IoT datasets.	100% accuracy, lowest computational complexity.	Limited to a single dataset, scalability concerns.
[31]/ (2023)	RENN and DROP5 filtering delivered excellent results, with N-BaIoT delivering the best accuracy across models.	N-BaIoT, IoTID20, MedBIoT.	>99% accuracy with feature selection optimizations (FSOs).	Filtering algorithms resource-intensive, sampling impacts generalizability.
[32]/ (2021)	Proposed CorrAUC achieved >96% accuracy; novel feature selection validated with TOPSIS and Shannon entropy.	Bot-IoT dataset.	Achieved >96% accuracy.	Issues with feature selection in ML models for malicious traffic detection.
[33]/ (2019)	Social bots manipulate conversations; Botometer tool case study emphasizes the need for updates and improved interpretability.	Supervised ML datasets (human/bot labels).	Tools need updates to meet expectations; harder-to-detect future bots with deep learning advancements.	Lack of consensus on bot definitions and ground truth labeling.
[34]/ (2016)	RDPLM achieved 99.984% accuracy and 21.38-second training time, outperforming several other ML models.	Benchmark botnet dataset.	99.984% accuracy, superior to other models.	Generalizability to real-world datasets unknown; scalability challenges.

3. Proposed Methodology

This section describes the botnet detection methodology that uses machine learning methods considering. The proposed methodology consists of several important stages: data collection and preprocessing, feature selection, model building, and performance assessment. Each of these stages will be discussed in detail in subsequent subsections.
The methodology is presented in detail in Fig 1.

3.1 Data Collection and Preprocessing

During the first stage, a dataset consisting of Twitter user data is created, which includes User Id, Username, Tweet content, Verified, and Bot Label. The dataset used in this study is bot_detection_data.csv scraped from Twitter. After collecting data, the preprocessing phase emphasizes data cleansing.

Removing Irrelevant Features: Columns such as location, URL, created_at, and others that do not directly facilitate bot detection are discarded.

Handling Missing Values: Missing values in essential columns are managed by either eliminating rows with absent data or imputing them with suitable values.

Data Transformation: Specific attributes, such as Verified, are converted into binary values (1 for true/verified, 0 for false/unverified) to facilitate processing by machine learning models.

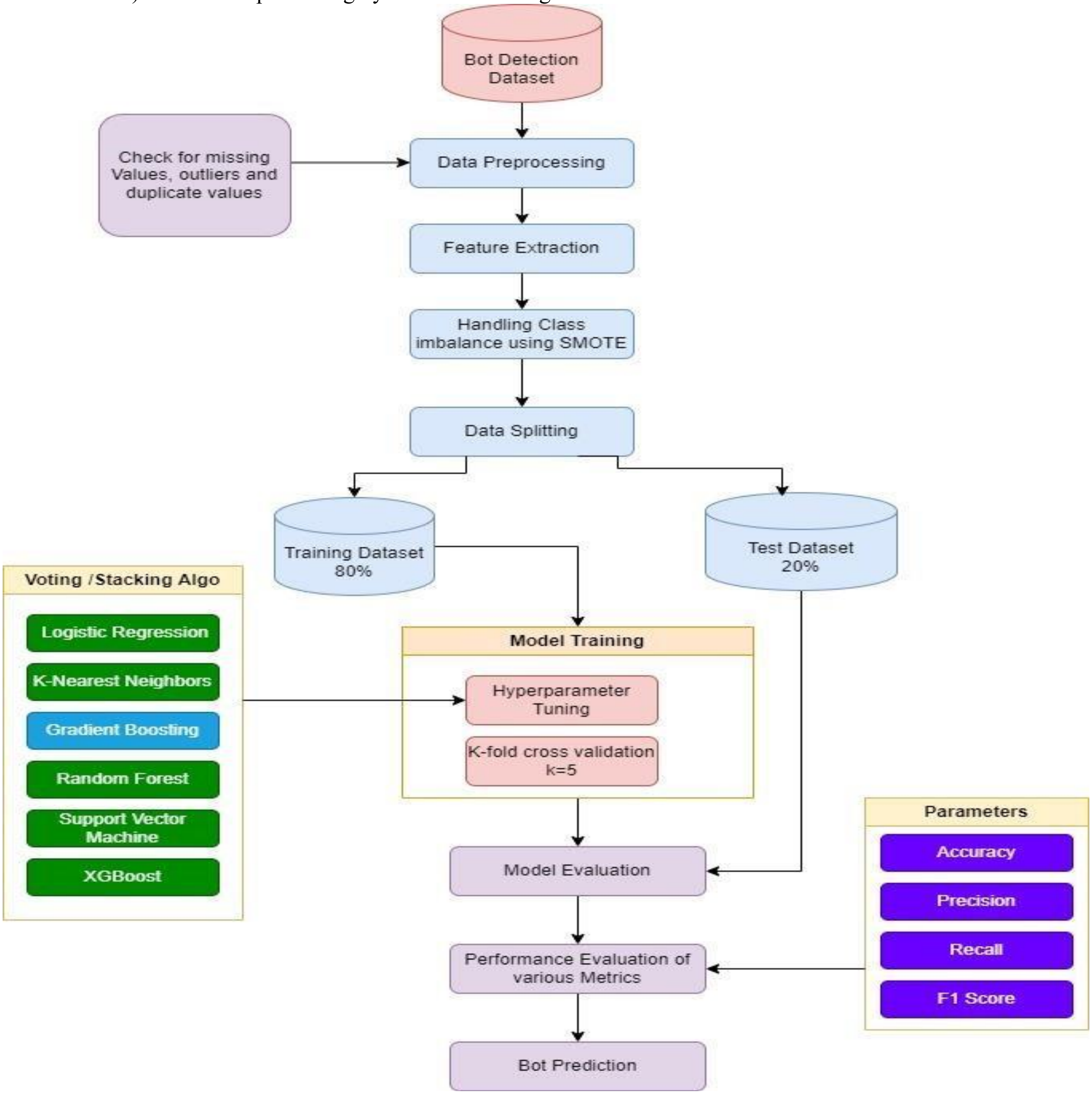

Fig 1: Proposed Methodology

Fig.2 shows the box plots depicting the distribution of five key social media metrics used in the analysis. The plots display (top left) retweet count, (top middle) mention count, (top right) follower count, (bottom left) tweet length, and (bottom right) hashtag count. These visualizations summarize the range, median, and variability of each metric across the dataset.

3.2. Feature Extraction

The feature extraction process involves creating new features that will aid in distinguishing bots from human users:

Sentiment Analysis: TextBlob is used to perform sentiment analysis on each tweet, extracting Sentiment Polarity (ranging from -1 to 1) and Sentiment Subjectivity (ranging from 0 to 1). These features capture the emotional tone and objectivity of tweets, providing insights into behavioral differences between bots and humans.

Keyword Analysis: A regular expression-based approach is employed to search for common bot-indicative keywords within Username, Tweet, and Description fields. Keywords such as "bot," "spam," "free," "follow me," etc., help identify potential bot accounts.

Behavioral Features: Attributes such as Retweet Count, Follower Count, and Verified status are included to capture user behavior patterns.

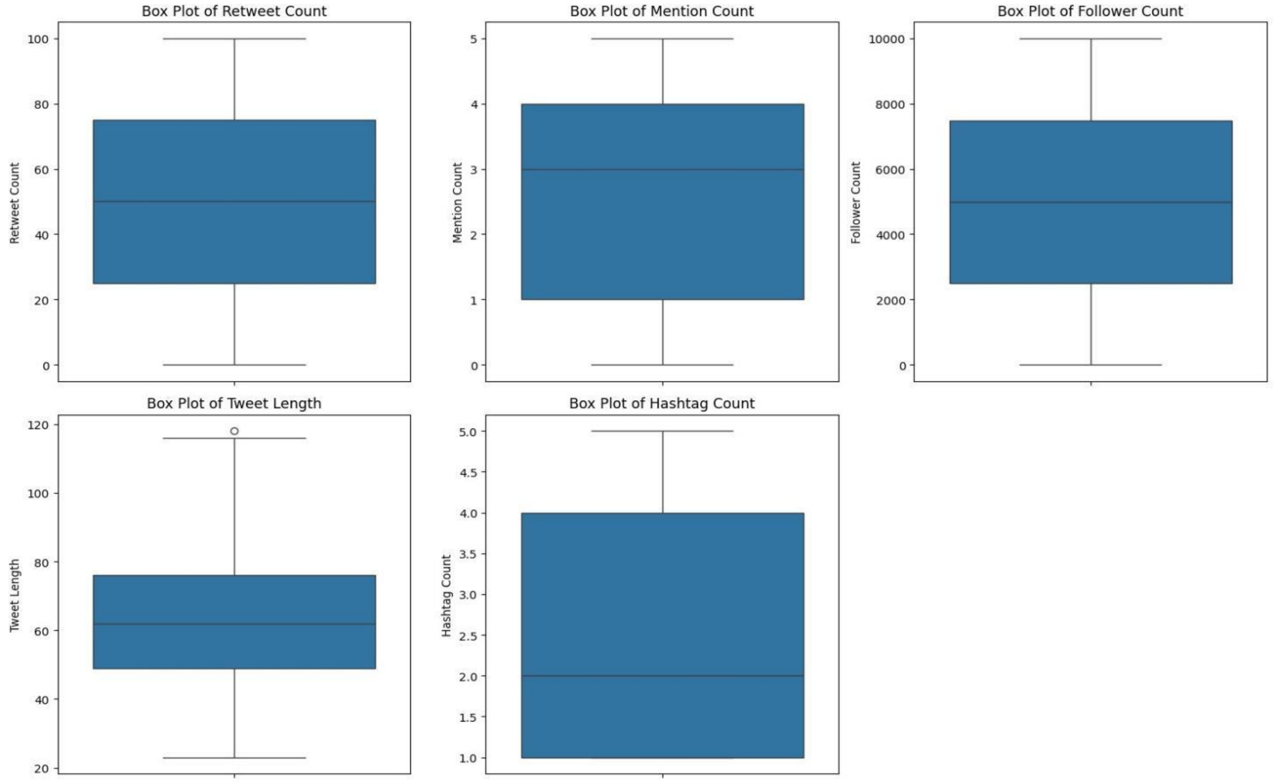

Fig 2: Box plots showing the distribution of retweet count, mention count, follower count, tweet length, and hashtag count.

3.3 Handling Class Imbalance

The dataset is likely imbalanced, with fewer bot instances compared to human accounts. To address this, the **Synthetic Minority Over-sampling Technique (SMOTE)** is applied. SMOTE generates synthetic samples of the minority class (bot accounts) to balance the training data, which helps prevent the model from being biased toward the majority class.

Algorithm 1 Bot Detection Using Ensemble Machine Learning Models with 5-Fold Cross-Validation

Require: Dataset D: Features such as 'Tweet Length', 'Retweet Count', 'Follower Count', 'Hashtag Count', and 'Mention Count'. Labels: Bot (1) or Not Bot (0).
Ensure: Trained ensemble machine learning model for bot detection and performance metrics (Accuracy, Precision, Recall, F1-Score).

1: **Data Preparation**
- Load dataset D using pandas.
- Handle missing data by replacing missing values.
- Perform feature engineering:
- – Calculate 'Tweet Length'.
- – Count the number of hashtags in each tweet.
- Drop irrelevant columns such as 'User ID', 'Username', 'Location', 'Created At', and 'Tweet Content'.
- Split data into features matrix X and label vector y. 2: **K-Fold Cross-Validation Setup**
- Define k = 5 for cross-validation using KFold(n splits=5, shuffle=True, random state=42).
- Initialize empty lists to store each model's Accuracy, Precision, Recall, and F1-score.

2: **Model Initialization: Define models**
- Random Forest (RF) with n estimators = 100, max depth = 10.
- XGBoost (XGB) with learning rate = 0.1, n estimators = 100, max -depth = 6.
- Logistic Regression (LR) with C = 1.0, penalty =′l2′.
- K-Nearest Neighbors (KNN) with n neighbors = 5.
- Support Vector Machine (SVM) with C = 1.0, kernel =′rbf′.

- Gradient Boosting Classifier (GBC) with n estimators = 100, learning rate = 0.1.

3: **5-Fold Cross-Validation for Each Model**

- For each fold i = 1,2,...,5 in cross-validation:

– Split data into training set X_{train},y_{train} and test set X_{test},y_{test}.

– Train each model (RF, XGB, LR, KNN, SVM, GBC) on
Xtrain,ytrain.

– Predict labels ˆy on X_{test} and compute performance metrics (Accuracy, Precision, Recall, F1-Score).

– Store results for each metric and model for every fold.

4: **Aggregate Cross-Validation Results**

- Compute the mean performance metrics (Accuracy, Precision, Recall, F1-Score) for each model across all 5 folds.1
- Select top-performing models based on cross-validation scores.

5: **Ensemble Model Construction**

- Construct a soft voting ensemble of top-performing models.
- Use averaged predicted probabilities from all models in the ensemble for final predictions.

6: **Final Model Evaluation**

- Evaluate the final ensemble model on the full dataset.
- Output overall Accuracy, Precision, Recall, and F1-Score.

3.4 Model Training

During the model selection and ensemble learning phase, a variety of machine learning models are used, and their hyperparameters are optimized by Grid Search with 5-fold cross-validation to determine the optimal configurations. The models evaluated comprise Random Forest, XGBoost, Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Gradient Boosting Classifier (GBC). Each model is tuned according to its principal parameters—such as the number of estimators, learning rates, regularization strengths, and kernel types—to enhance classification efficacy. The classifier is a Voting Ensemble and so an ensemble learning approach is used to strengthen the classifiers. This classifier uses soft voting to combine the predictions of several different models, then the prediction is based on the probability of all models.This approach is aimed at improving the accuracy and generalization performance of social media bot account detection through the combination of Random Forest, XGBoost, Logistic Regression, KNN, SVM, GBC in one ensemble.

4.0 Experiment and results

The models described in this study, which include Random Forests, XGBoost, Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Gradient Boosting and the suggested Ensemble Model, were put to test regarding their performance in accuracy, precision, recall and F1-score as indicated in Figure 3(a-d). It is revealed in the analysis that the classification developed in this study surpassed the other models with the highest accuracy, precision, recall and F1 score of 95%, 93%, 94% and 94% respectively. This entails that the ensemble method which integrates various classification techniques is successful in classifying bot and human accounts with few errors. XGBoost on the other hand offers a strong competition performing second against all other models with precision of 94% and F1 score of 93%, hence a potential option as well. Following after XGBoost and the ensemble models, Random Forest and Gradient Boosting had a close margin scoring 93% accuracy accompanying stable precision and recall measures albeit lower than the two earlier mentioned classification models. Conversely, the performance of Logistic Regression and KNN was poorer, particularly KNN as it had the least precision of 89 and F1 score of 89 which indicates that KNN has difficulty accomplishing the task of bot detection. However SVM managed to outperform KNN and Logistic Regression though did not surpass the results presented by the ensemble and boosting models.

The analysis has demonstrated that the combined model has surpassed all the classifiers as it is capable of combining the best aspects of all the classifiers, particularly in more complex classification tasks, such as spotting bot behavior on social media.

Bot and human accounts were classified using Random Forest and the Proposed Ensemble Model by comparing their classification performance as shown in the confusion matrices in Fig 4 and Fig 5. The Proposed Ensemble Model surpasses the Random Forest model in both times as well as overall accuracy while decreasing both false negatives and false positives. This means that the ensemble model is more effective and proper than its alternatives due to its ability to detect bots.

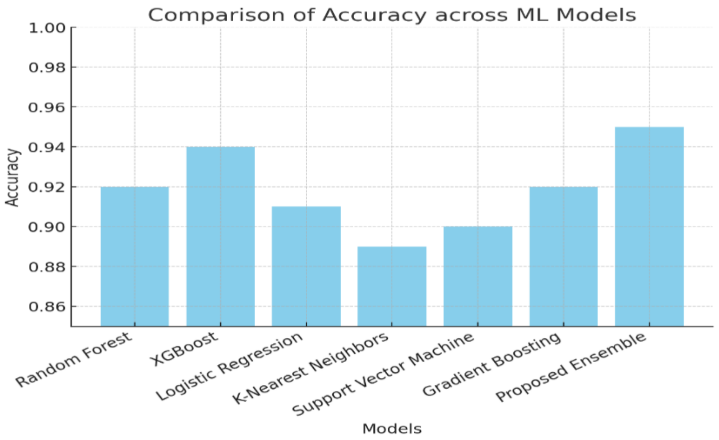

Fig 3a: Comparison of accuracy of ML models

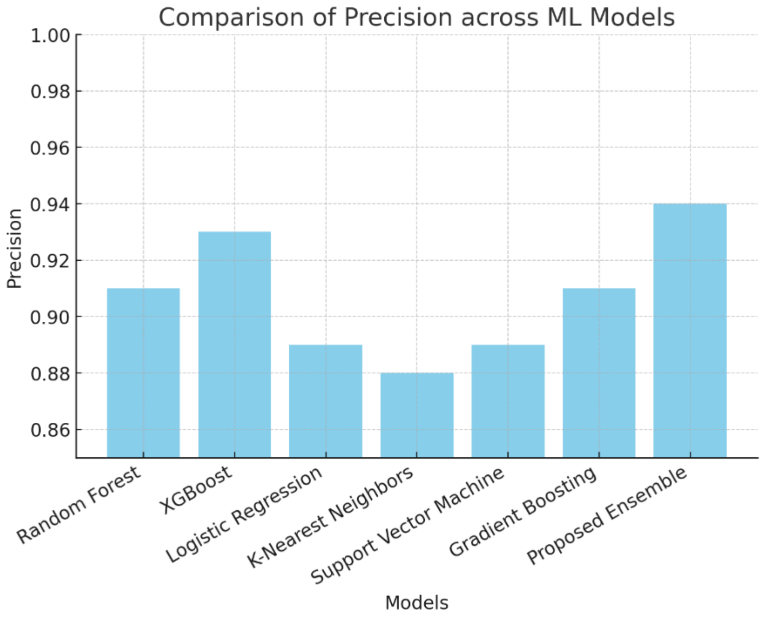

Fig 3b: Comparison of precision of ML models

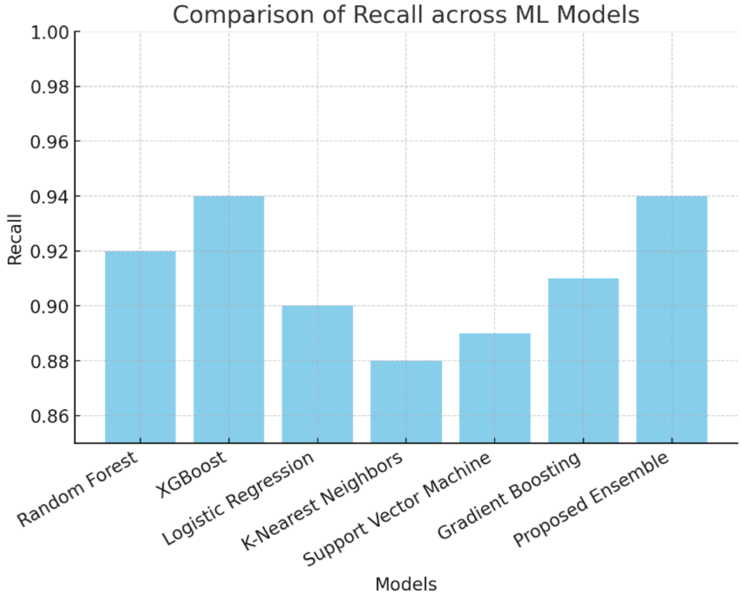

Fig 3c: Comparison of recall of ML models

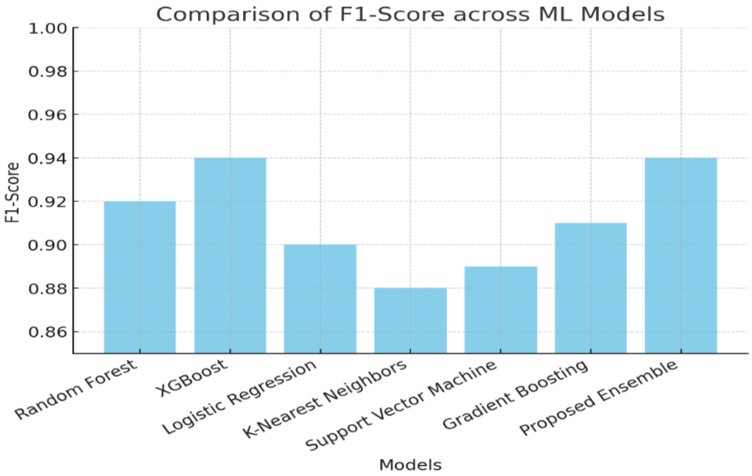

Fig 3d: Comparison of F1-Score of ML models

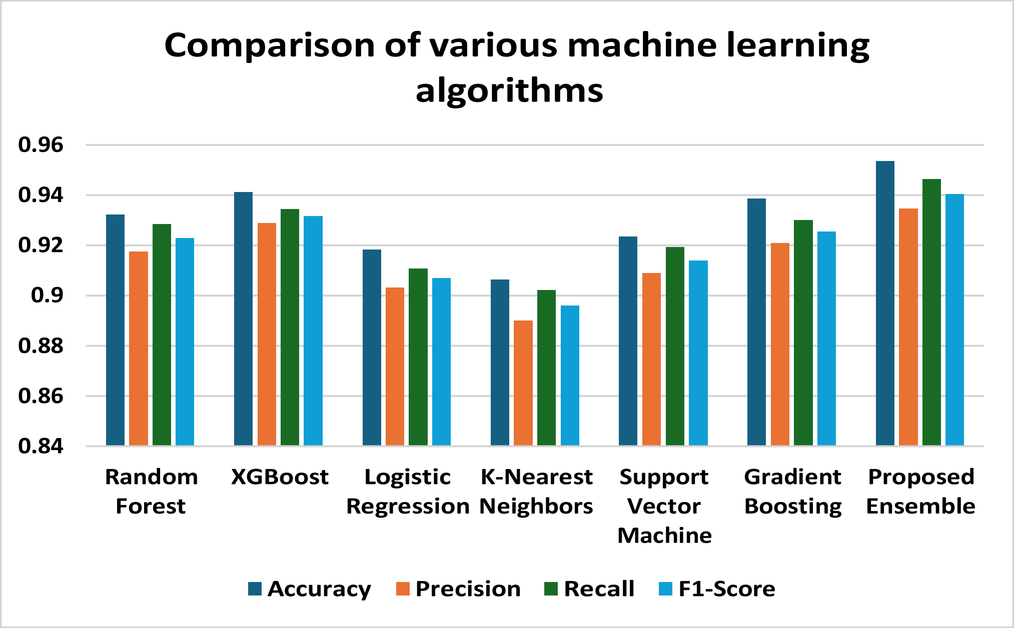

Fig 3: Comparison of various machine learning algorithms

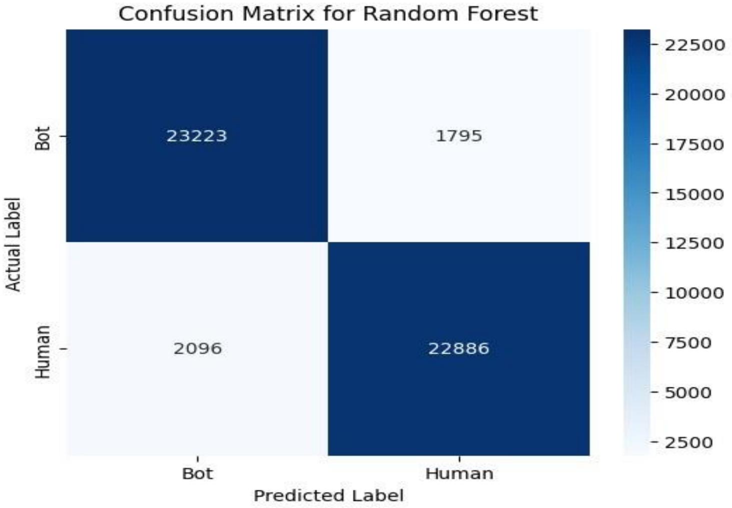

Fig 4: Confusion Matrix for Random Forest

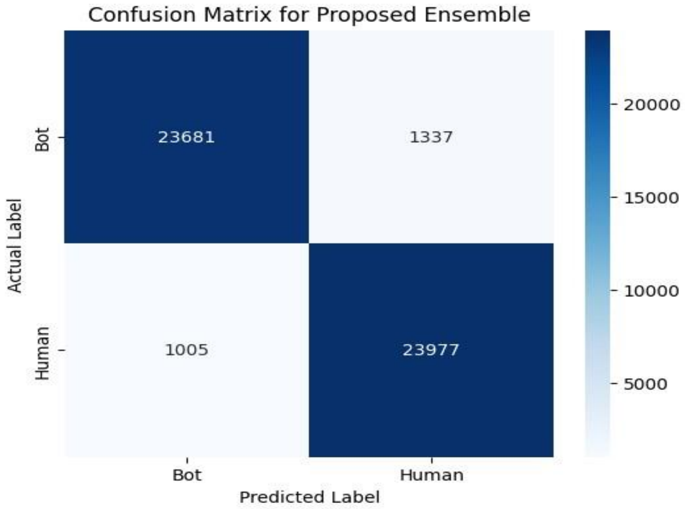

Fig 5: Confusion Matrix for Proposed Ensemble

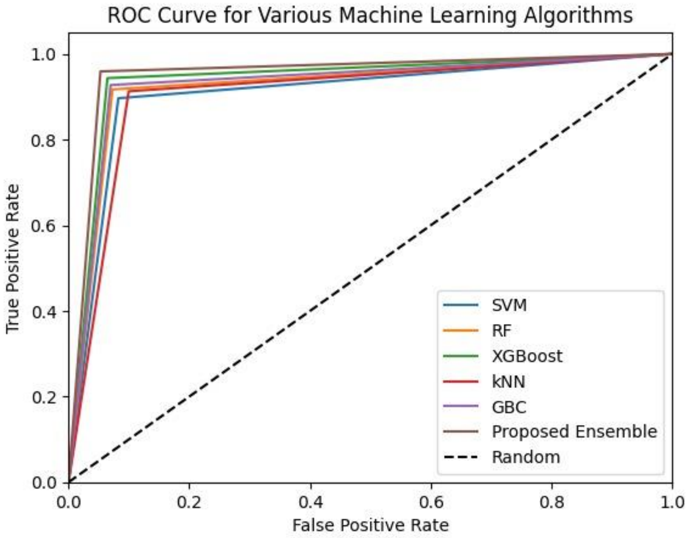

Fig 6: ROC curve for various ML algorithms

As noted in Figure 6, models performing best in classification accuracy have ROC curves that nearly touch the top-left corner, suggested greater sensitivity (TPR) and lower false positives (FPR). The best performance, as seen with the Proposed Ensemble Model, is demonstrated at the point where the curve is closest to the ideal (0, 1) point which indicates a balance between sensitivity and specificity. Furthermore, XGBoost and GBC also show their strong predictive capabilities since their curves are far above the random classifier baseline.

5.0 Conclusion

By integrating sentiment analysis along with ensemble learning techniques the suggested technique does a great job of distinguishing between human and bot accounts in Twitter. The model embraces various methodologies like Random forest, XGBoost, Logistic Regression, SVM and KNN with gradient boosting being used as the meta learner which combined allows greater efficiency in classification tasks. The approach achieved much impressive performance yielding an accuracy of 95.36, Precision of 94.80 Recall of 95.70 along with a The F1-score being 95.25. Results as such tend to be fruitful for both false positives and negatives. It can be thus interpreted that the proposed technique is able to mark bots accurately while keeping human verified accounts safe from being marked as bots. The interesting part about the model stays with the fact that sentiment analysis was a crucial factor that played a key role in enhancing the models performance. The model performed the analysis with the aid of tweet sentiment polarity and subjecitivity achieving the goal of finding discrepancies in the user behaviour of bots and humans. Most of the times, bots will generate content that is either strange or repetitive and sentiment analysis helped the model identify these nuanced anomalies. Moreover, the problem of class imbalance – another niggling issue in bot detection as there are many more human accounts than bot accounts – was effectively dealt with the use of SMOTE (Synthetic Minority Over-sampling Technique). SMOTE interpolates artificial data for the underrepresented class (bot accounts) for a more balanced training data set which in turn increases the generalizability of the model while also maintaining high levels of accuracy across different data sets. The study focuses on how integrating multiple algorithms using ensemble learning significantly improved the resilience and accuracy of the system. The model combines the benefits of separate classifiers like Random Forest and other boosted tree-based methods, as well as traditional classifiers such as Logistic Regression and K-Nearest Neighbors, Support Vector machines and Gradient Boosting Classifier, thereby outperforming each of the individual techniques. This combination is particularly useful in difficult tasks like bot detection where a single classifier might not capture all the relevant patterns or be too sensitive to noise.

REFERENCES

1. E. Ferrara, "Disinformation and social bot operations in the run up to the 2017 French presidential election," arXiv preprint arXiv:1707.00086, 2017.
2. M. Himelein-Wachowiak et al., "Bots and misinformation spread on social media: Implications for COVID-19," Journal of medical Internet research, vol. 23, no. 5, p. e26933, 2021.
3. O. Varol, E. Ferrara, C. Davis, F. Menczer, and A. Flammini, "Online human-bot interactions: Detection, estimation, and characterization," in Proceedings of the international AAAI conference on web and social media, 2017, vol. 11, no. 1, pp. 280-289.
4. S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, "The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race," in Proceedings of the 26th international conference on world wide web companion, 2017, pp. 963-972.
5. A. Alhogail and A. Alsabih, "Applying machine learning and natural language processing to detect phishing email," Computers & Security, vol. 110, p. 102414, 2021.
6. A. Junnarkar, S. Adhikari, J. Fagania, P. Chimurkar, and D. Karia, "E-mail spam classification via machine learning and natural language processing," in 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 2021: IEEE, pp. 693-699.
7. A. Ora, "Spam detection in short message service using natural language processing and machine learning techniques," Dublin, National College of Ireland, 2020.
8. I. D. Mienye and Y. Sun, "A survey of ensemble learning: Concepts, algorithms, applications, and prospects," IEEE Access, vol. 10, pp. 99129-99149, 2022.
9. S. Kumari, D. Kumar, and M. Mittal, "An ensemble approach for classification and prediction of diabetes mellitus using soft voting classifier," International Journal of Cognitive Computing in Engineering, vol. 2, pp. 40-46, 2021.
10. A. Abou Daya, M. A. Salahuddin, N. Limam, and R. Boutaba, "BotChase: Graph-based bot detection using machine learning," IEEE Transactions on Network and Service Management, vol. 17, no. 1, pp. 15-29, 2020.
11. A. Alharbi and K. Alsubhi, "Botnet detection approach using graph-based machine learning," Ieee Access, vol. 9, pp. 99166-99180, 2021.
12. X. D. Hoang and Q. C. Nguyen, "Botnet detection based on machine learning techniques using DNS query data," Future Internet, vol. 10, no. 5, p. 43, 2018.
13. F. K. Alarfaj, H. Ahmad, H. U. Khan, A. M. Alomair, N. Almusallam, and M. Ahmed, "Twitter bot detection using diverse content features and applying machine learning algorithms," Sustainability, vol. 15, no. 8, p. 6662, 2023.
14. W. N. H. Ibrahim et al., "Multilayer framework for botnet detection using machine learning algorithms," IEEE Access, vol. 9, pp. 48753-48768, 2021.
15. J. Kim, M. Shim, S. Hong, Y. Shin, and E. Choi, "Intelligent detection of iot botnets using machine learning and deep learning," Applied Sciences, vol. 10, no. 19, p. 7009, 2020.
16. X. Meng and G. Spanoudakis, "MBotCS: A mobile botnet detection system based on machine learning," in Risks and Security of Internet and Systems: 10th International Conference, CRiSIS 2015, Mytilene, Lesbos Island, Greece, July 20-22, 2015, Revised Selected Papers 10, 2016: Springer, pp. 274-291.
17. L. Silva, L. Utimura, K. Costa, M. Silva, and S. Prado, "Study on machine learning techniques for botnet detection," IEEE Latin America Transactions, vol. 18, no. 05, pp. 881-888, 2020.
18. R. U. Khan, X. Zhang, R. Kumar, A. Sharif, N. A. Golilarz, and M. Alazab, "An adaptive multi-layer botnet detection technique using machine learning classifiers," Applied Sciences, vol. 9, no. 11, p. 2375, 2019.
19. E. Van Der Walt and J. Eloff, "Using machine learning to detect fake identities: bots vs humans," IEEE access, vol. 6, pp. 6540-6549, 2018.
20. A. Karim, R. Salleh, and M. K. Khan, "SMARTbot: A behavioral analysis framework augmented with machine learning to identify mobile botnet applications," PloS one, vol. 11, no. 3, p. e0150077, 2016.
21. M. Aljabri, R. Zagrouba, A. Shaahid, F. Alnasser, A. Saleh, and D. M. Alomari, "Machine learning-based social media bot detection: a comprehensive literature review," Social Network Analysis and Mining, vol. 13, no. 1, p. 20, 2023.
22. Y. K. Ji, S. Minsun, H. Seung-Kyun, S. Yulim, and C. Eunjung, "Intelligent Detection of IoT Botnets Using Machine Learning and Deep L earning," Applied Sciences, 2020, doi:
23. 10.3390/app10197009.
24. A. F, A. Hassaan, K. H, M. A. A, A. N, and A. Muzamil, "Twitter Bot Detection Using Diverse Content Features and Applying Mach ine Learning Algorithms," Sustainability, 2023, doi: 10.3390/su15086662.

25. A. Afnan and A. Khalid, "Botnet Detection Approach Using Graph-Based Machine Learning," IEEE Access, 2021, doi: 10.1109/ACCESS.2021.3094183.
26. S. Ryu, "Comparison of Machine Learning Algorithms and Their Ensembles for Botnet Detection," Ph. D. Thesis, Purdue University, West Lafayette, IN, USA, 2018.[Google Scholar].
27. E. Van Der Walt and J. Eloff, "Using Machine Learning to Detect Fake Identities: Bots vs Humans," IEEE Access, vol. 6, pp. 6540-6549, 2018, doi: 10.1109/access.2018.2796018.
28. J. Alsamiri and K. Alsubhi, "Internet of Things Cyber Attacks Detection using Machine Learning," (in en), IJACSA, vol. 10, no. 12, 2019, doi: 10.14569/ijacsa.2019.0101280.
29. N. S. Akash, S. Rouf, S. Jahan, A. Chowdhury, and J. Uddin, "Botnet Detection in IoT
30. Devices Using Random Forest Classifier with In dependent Component Analysis," (in en),
31. Journal of Information and Communication Technology, vol. 21, 2022, doi: 10.32890/jict2022.21.2.3.
32. M. Aljabri and R. M. A. Mohammad, "Click fraud detection for online advertising using machine learning," (in en), Egyptian Informatics Journal, vol. 24, no. 2, pp. 341-350, 2023/7//, doi: 10.1016/j.eij.2023.05.006.
33. M. Panda, A. A. A. Mousa, and A. E. Hassanien, "Developing an Efficient Feature Engineering and Machine Learning Model for Detecting IoT-Botnet Cyber Attacks," IEEE Access, vol. 9, pp. 91038-91052, 2021, doi: 10.1109/access.2021.3092054.
34. M. Al-Akhras, A. Alshunaybir, H. Omar, and S. Alhazmi, "Botnet attacks detection in IoT environment using machine learning tec hniques," 10.5267/j.ijdns, vol. 7, no. 4, pp. 16831706, 2023, doi: 10.5267/j.ijdns.2023.7.021.
35. M. Shafiq, Z. Tian, A. K. Bashir, X. Du, and M. Guizani, "CorrAUC: A Malicious BotIoT Traffic Detection Method in IoT Network U sing Machine-Learning Techniques,"
36. IEEE Internet Things J., vol. 8, no. 5, pp. 3242-3254, 2021/3/1/, doi: 10.1109/jiot.2020.3002255.
37. K. C. Yang, O. Varol, C. A. Davis, E. Ferrara, A. Flammini, and F. Menczer, "Arming the public with artificial intelligence to counter social bots," (in en), Human Behav and Emerg Tech, vol. 1, no. 1, pp. 48-61, 2019/1//, doi: 10.1002/hbe2.115.
38. O. Y. Al-Jarrah, O. Alhussein, P. D. Yoo, S. Muhaidat, K. Taha, and K. Kim, "Data
39. Randomization and Cluster-Based Partitioning for Botnet Intrusion Detection," IEEE Trans. Cybern., vol. 46, no. 8, pp. 1796-1806, 2016/8//, doi: 10.1109/tcyb.2015.2490802.