Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Grey Wolf Optimizer-Based Feature Selection with Mutual Information Refinement for Machine Learning-Based Bot Detection on Social Media

Aditya Vardhan

aditya202202@dei.ac.in

Dayalbagh Educational Institute

Mohit Yadav

Dayalbagh Educational Institute

Amarjeet Singh Chauhan

Dayalbagh Educational Institute

Sanjay Saini

Dayalbagh Educational Institute

Article

Keywords: Social Media Bot Detection, Grey Wolf Optimizer, Mutual Information, Feature Selection, Machine Learning, XGBoost

Posted Date: August 27th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10580582/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Grey Wolf Optimizer-Based Feature Selection with Mutual Information Refinement for Machine Learning-Based Bot Detection on Social Media

Aditya Vardhan (Corresponding author) – vardhana3098@gmail.com

Mohit Yadav- mohityadavdei@yahoo.com

Amarjeet Singh Chauhan- amarjeetsinghchauhan96@gmail.com

Sanjay Saini- Sanjay.saini@dei.ac.in

Department of Physics and Computer Science, Dayalbagh Educational Institute, Agra-282005, INDIA

Abstract: Social media bots are increasingly used to spread spam, misinformation, and malicious content, making accurate bot detection a significant cybersecurity challenge. This study proposes a **Grey Wolf Optimizer with Mutual Information (GWO–MI)** based feature selection framework to improve machine learning-based bot detection. The proposed approach first applies Grey Wolf Optimizer to identify informative features and then employs Mutual Information to refine the selected subset. Experiments were conducted on a publicly available Twitter bot detection dataset containing **8,386 accounts** and **68 features**. Five machine learning classifiers, including Random Forest, Decision Tree, Logistic Regression, Support Vector Machine, and XGBoost, were evaluated using different feature subset sizes. The proposed framework reduced the original feature space from **68 to 25 features**, with further refinement through Mutual Information. Among the evaluated models, **XGBoost achieved the best classification accuracy of 98.87%**. The results demonstrate that the proposed framework provides effective feature reduction while maintaining competitive bot detection performance.

Keywords: Social Media Bot Detection, Grey Wolf Optimizer, Mutual Information, Feature Selection, Machine Learning, XGBoost.

1. Introduction

Social media platforms have become one of the primary channels for communication, information sharing, marketing, and public engagement. Platforms such as X (formerly Twitter), Facebook, Instagram, and LinkedIn generate enormous volumes of user-generated content every day and influence social, political, and economic activities worldwide (Aljabri et al. 2023). While these platforms provide numerous benefits by enabling rapid information dissemination and global connectivity, they have also become attractive targets for malicious activities that threaten the reliability and security of online communities. Among these challenges, the increasing presence of automated social media accounts, commonly referred to as **social bots**, has emerged as a major concern. Social bots are software-controlled accounts capable of automatically publishing content, interacting with users, and imitating human behaviour (Abou Daya et al. 2019). Although some bots are developed for legitimate purposes, such as customer support, news broadcasting, and automated notifications, malicious bots are frequently employed to spread spam, misinformation, phishing campaigns, fake reviews, and coordinated propaganda. Such activities not only influence public opinion but also compromise the credibility and trustworthiness of social media platforms (Hayawi et al. 2023).

The detection of malicious bot accounts has therefore become an active research area in machine learning and cybersecurity. Traditional machine learning algorithms, including Random Forest, Decision Tree, Logistic Regression, and Support Vector Machine, have demonstrated promising performance for bot classification. However, the effectiveness of these algorithms largely depends on the quality of the selected input features (Efthimion et al. 2018). Real-world social media datasets often contain redundant, noisy, and highly correlated attributes, which increase computational complexity and may negatively affect classification performance. Consequently, feature selection has become an essential preprocessing step for identifying the most informative features while reducing the dimensionality of the dataset.

2. Literature Survey

The rapid growth of social media platforms has significantly increased the influence of automated accounts, commonly known as social bots. While some bots perform legitimate tasks such as customer support and information dissemination, malicious bots manipulate public opinion, spread misinformation, amplify political propaganda, conduct phishing attacks, and influence financial markets. Consequently, the development of accurate and computationally efficient bot detection systems has become an active area of research. Recent studies have primarily focused on three interconnected research directions: feature selection, machine learning and deep learning models, and graph-based learning approaches.

One of the most important challenges in bot x detection is identifying the most informative features from high-dimensional social media datasets. Feature selection not only

reduces computational complexity but also improves classification accuracy by eliminating redundant and irrelevant attributes. Wang et al. (2019) proposed an input feature selection method based on feature set equivalence and mutual information gain maximization. Their approach effectively preserved the discriminative power of selected features while minimizing redundancy, demonstrating that optimized feature subsets significantly enhance classification performance. Similarly, Li et al. (2017) provided a comprehensive overview of recent advances in feature selection methods, categorizing techniques into filter, wrapper, and embedded approaches while highlighting their applications across various machine learning domains. Dhal and Azad (2022) further extended this discussion by surveying feature selection techniques employed in healthcare, cybersecurity, image processing, and data mining, concluding that hybrid feature selection methods generally outperform individual algorithms due to their ability to balance exploration and exploitation during optimization.

The application of feature selection has been particularly significant in Twitter bot detection. Abreu et al. (2020) demonstrated that reducing the number of input features could maintain competitive detection accuracy while substantially lowering computational cost. Their work highlighted that carefully selected metadata attributes are often sufficient for distinguishing human users from automated accounts. Building upon this concept, Alothali et al. (2021) introduced a hybrid feature selection framework that combines filter-based and wrapper-based methods to identify optimal Twitter profile metadata. Their results showed improved detection performance while simultaneously reducing model complexity, emphasizing the importance of intelligent feature engineering before classification.

Machine learning has traditionally formed the foundation of bot detection research. Yang et al. (2019) investigated how artificial intelligence can empower ordinary users to identify malicious social bots, emphasizing interpretable machine learning models capable of assisting public decision-making rather than solely relying on automated moderation systems. Dimitriadis et al. (2021) proposed Social Botomics, an explainable ensemble machine learning framework that integrates multiple classifiers to perform multiclass bot detection. Their model demonstrated that ensemble learning not only improves predictive performance but also provides greater transparency regarding feature importance and classification decisions, an increasingly important requirement for trustworthy artificial intelligence.

As social media behavior becomes increasingly dynamic, researchers have shifted toward adaptive learning frameworks capable of handling evolving bot behaviors. Alothali et al. (2023) introduced SEBD, a Stream Evolving Bot Detection framework that employs Probably Approximately Correct (PAC) learning to maintain classification accuracy and confidence levels over continuously evolving data streams. Unlike conventional static models, SEBD dynamically updates its learning process as new behavioral patterns emerge, addressing one of the primary limitations of traditional supervised learning techniques.

Recent years have witnessed the rapid adoption of deep learning techniques for bot detection due to their ability to automatically extract complex behavioral patterns from heterogeneous data. Martin-Gutierrez et al. (2021) proposed a transformer-based deep learning architecture capable of learning contextual relationships within Twitter data. Their model demonstrated robust performance across multiple benchmark datasets, outperforming several conventional machine learning approaches. Similarly, Sengar et al. (2020) presented a multilayered deep learning framework for social network bot detection, showing that hierarchical neural architectures effectively capture nonlinear relationships among user profile, network, and behavioral features.

Graph-based learning has emerged as another promising direction because social bots typically operate within coordinated networks rather than as isolated accounts. Daya et al. (2019) introduced a graph-based machine learning approach that models relationships among users through graph representations before applying classification algorithms. Their subsequent work, BotChase (2020), extended this framework by incorporating graph analytics with machine learning techniques, significantly improving the identification of coordinated bot communities. These studies demonstrated that exploiting network topology provides valuable complementary information beyond individual user metadata.

More recently, graph neural networks have further advanced this research area. Alothali et al. (2022) proposed Bot-MGAT, a transfer learning framework based on a Multi-view Graph Attention Network (MGAT). By integrating multiple graph views with attention mechanisms, their model effectively captured diverse interaction patterns between accounts and demonstrated superior generalization across different datasets. Similarly, Ye et al. (2023) introduced HOFA (Homophily-Oriented Augmentation and Frequency Adaptive Attention), which combines graph augmentation with adaptive attention mechanisms to improve robustness against increasingly sophisticated bot behaviors. These graph attention models represent the current state-of-the-art in exploiting relational information for bot detection.

Another emerging research direction involves user profiling techniques. Heidari et al. (2022) proposed an online user profiling framework that integrates behavioral characteristics, linguistic features, temporal activity patterns, and account metadata to improve Twitter bot detection. Their study demonstrated that combining multiple behavioral dimensions produces more reliable predictions than relying solely on profile metadata or textual features.

While supervised learning dominates the literature, semi-supervised and unsupervised learning have gained increasing attention due to the scarcity of labeled datasets. Khalil et al. (2020) investigated feature selection strategies specifically designed for unsupervised bot detection, demonstrating that meaningful clustering can still be achieved without labeled training data when appropriate feature subsets are selected. Similarly, Mbona and Eloff (2023) developed a semi-supervised machine learning framework capable of classifying social media bots as malicious or benign using limited labeled samples. Their results suggest that semi-supervised approaches provide a practical solution for real-world environments where obtaining comprehensive annotations is expensive and time-consuming.

The availability of high-quality benchmark datasets has also contributed significantly to research progress. Cea (2021) released a publicly available supervised Twitter bot detection dataset through Zenodo, enabling standardized evaluation and reproducibility across different machine learning models. Such benchmark datasets have become essential for fair comparison among emerging detection algorithms.

Beyond bot detection, several studies have demonstrated the broader effectiveness of hybrid artificial intelligence frameworks in cybersecurity applications. Katoch et al. (2021) reviewed the evolution of genetic algorithms and highlighted their widespread integration with machine learning for optimization problems, including feature selection. Remmide et al. (2024) proposed a hybrid framework combining deep learning with case-based reasoning for phishing email detection, achieving improved adaptability and classification performance. Subsequently, Remmide et al. (2025) introduced a privacy-preserving federated deep learning framework for smishing attack detection, demonstrating that distributed learning architectures can simultaneously enhance security, scalability, and user privacy.

3. Proposed Methodology

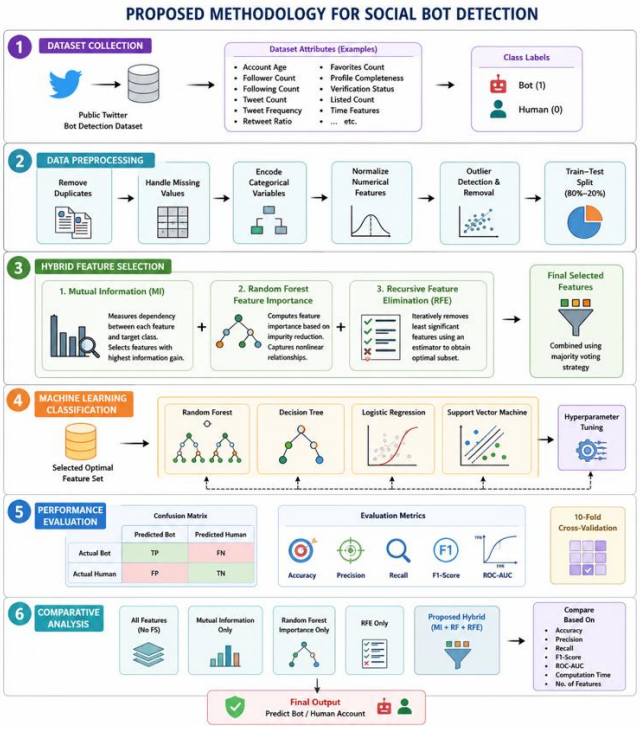


Figure 3.1. **Proposed methodology for social bot detection using a hybrid feature selection framework.** The framework integrates data preprocessing, hybrid feature selection (MI, RF, and RFE), machine learning classification, performance evaluation, and comparative analysis to accurately classify social media accounts as bots or humans.

3.1 Dataset and Preprocessing

The experiments were conducted using the publicly available **Dataset for Supervised Bot Detection on Twitter (v1.0)**, which contains **8,386 Twitter accounts**, including **4,912 bot accounts** and **3,474 genuine human accounts**. Each account is represented by **68 handcrafted features** that capture various aspects of user behavior and profile characteristics. These features are categorized into three groups: **Content Features (19)**, which describe tweet characteristics such as tweet length, hashtags, URLs, mentions, and retweet statistics; **Account Information Features (20)**, which represent profile metadata including verification status, profile settings, username characteristics, and account configuration; and **Account Usage Features (29)**, which characterize user activity, follower–friend relationships, interaction statistics, account age, and various ratio-based behavioral attributes. The target variable, **class_bot**, is a binary label indicating whether an account is a bot (1) or a genuine user (0).

Prior to model development, the dataset underwent a comprehensive preprocessing stage to ensure data quality and consistency. Missing values and duplicate records were identified and removed to eliminate incomplete and redundant observations (Araujo et al. 2023). The target variable was separated from the predictor variables, resulting in a feature matrix (**X**) and a target vector (**y**). Subsequently, the dataset was partitioned into training and testing subsets

using an **80:20 stratified train–test split**, producing **6,708 training samples** and **1,678 testing samples** while preserving the original class distribution. The pre-processed dataset was then used for feature selection using Genetic Algorithm (GA) (Velasco-Mata et al. 2023), Particle Swarm Optimization (PSO) (Narayan et al. 2021), and Grey Wolf Optimizer (GWO) (Shinan et al. 2022), followed by Mutual Information (MI)-based feature refinement and machine learning classification (Taher et al. 2023).

Table 3.1: Categories of Features in the Twitter Bot Detection Dataset

Content Features (19)	Account Information Features (20)	Account Usage Features (29)
statuses_count	id	followers_count
favourites_count	default_profile	friends_count
min_tweet_length	default_profile_image	listed_count
max_tweet_length	geo_enabled	num_reply
avg_tweet_length	profile_use_background_image	num_retweet
min_urls	profile_background_tile	min_favorite
max_urls	utc_offset	max_favorite
avg_urls	protected	avg_favorite
min_hashtags	verified	profile_pic_frec
max_hashtags	digits_name	friends_followers_ratio
avg_hashtags	digits_screen_name	friends_followers_ratio_beq_50
max_mentions	name_length	friends_followers_square_ratio
avg_mentions	screen_name_length	friends_followers+friends_ratio
max_retweets	screen_name_length_name_length_ratio	2_followers_minus_friends
avg_retweets	name_contains_bot	2_followers_beq_100
description_length	screen_name_contains_bot	lists_followers_ratio
description_contains_bot	name_entropy	retweet_followers_ratio
avg_urls_status_ratio	screen_name_entropy	favorites_followers_ratio
avg_mentions_status_ratio	name_screen_name_lev_sim	lists_status_ratio
	profile_has_url	retweet_status_ratio
		favorites_status_ratio
		reply_status_ratio
		avg_favorite_status_ratio
		account_age
		followers_account_age_ratio
		friends_account_age_ratio
		statuses_account_age_ratio
		favourites_account_age_ratio
		lists_account_age_ratio

3.2 Proposed GWO–MI Feature Selection Framework

The proposed framework aims to improve social media bot detection by combining the global search capability of the **Grey Wolf Optimizer (GWO)** with the statistical ranking ability of **Mutual Information (MI)** for feature selection. The overall objective is to identify a compact subset of highly informative features that maximizes classification performance while reducing computational complexity. Unlike conventional wrapper-based methods that directly evaluate the complete feature space, the proposed framework performs feature

selection in two sequential stages. Initially, the Grey Wolf Optimizer explores the search space and identifies an optimal subset of candidate features. Subsequently, Mutual Information is applied to rank the selected features according to their dependency with the target class, retaining only the most informative attributes for final classification (A. Guendouz et al. 2026s). The complete workflow of the proposed framework is illustrated in **Figure 3.2**, The process begins with the acquisition of the Twitter bot detection dataset, followed by data preprocessing, including missing value removal, duplicate elimination, and train-test partitioning. The preprocessed dataset is then provided as input to the Grey Wolf Optimizer, where each wolf represents a candidate feature subset encoded as a binary vector. During the optimization process, the population iteratively updates its positions based on the leadership hierarchy of the alpha, beta, and delta wolves until an optimal feature subset is obtained. The selected features are subsequently ranked using Mutual Information, and multiple feature subset sizes are evaluated to determine the optimal balance between feature dimensionality and classification performance. Finally, the refined feature subsets are classified using Random Forest, Decision Tree, Logistic Regression, Support Vector Machine, and Extreme Gradient Boosting (XGBoost), and their performances are compared using Accuracy, Precision, Recall, and F1-score.

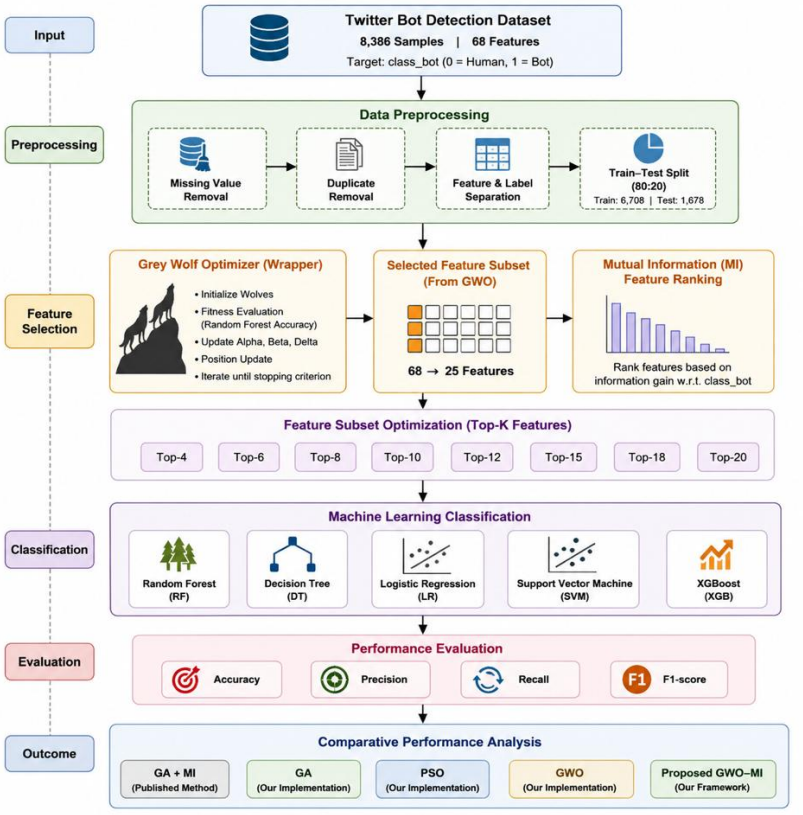


Figure 3.2: Overall architecture of the proposed GWO–MI feature selection framework for machine learning-based social media bot detection.

3.2.1 Mathematical Formulation

Let

- Dataset

$$D = \{X, Y\}$$

where

- $X = \{x_1, x_2, \ldots, x_n\}$ represents the feature matrix.
- Y denotes the class labels.

Each wolf is represented as a binary feature vector:

$$W_i = [w_1, w_2, \ldots, w_n]$$

where

$$w_j = \begin{cases} 1, & \text{feature selected} \\ 0, & \text{otherwise} \end{cases}$$

The fitness of each wolf is evaluated using Random Forest classification accuracy:

$$Fitness(W_i) = Accuracy(X_{selected}, Y)$$

The position update mechanism of GWO is defined as

$$\vec{D} = | \vec{C} \cdot \vec{X}_p - \vec{X} |$$

$$\vec{X}(t+1) = \vec{X}_p - \vec{A} \cdot \vec{D}$$

where

$$\vec{A} = 2a\vec{r_1} - a$$

$$\vec{C} = 2\vec{r_2}$$

and

$$a = 2 - \frac{2t}{T}$$

where T is the maximum number of iterations.

After optimization, Mutual Information ranks the selected features according to

$$MI(X,Y) = \sum_{x,y} P(x,y) \log \frac{P(x,y)}{P(x)P(y)}$$

The top-ranked features are finally used for classification.

3.3 Grey Wolf Optimizer (GWO)

Grey Wolf Optimizer (GWO) is a population-based metaheuristic optimization algorithm inspired by the social hierarchy and cooperative hunting behavior of grey wolves. In the optimization process, each wolf represents a candidate solution, where a binary vector encodes the selected feature subset. The population is organized into four hierarchical levels: **Alpha (α)**, **Beta (β)**, **Delta (δ)**, and **Omega (ω)**. The alpha wolf denotes the best solution, followed by beta and delta wolves, while the remaining omega wolves update their positions according to these three leaders. This leadership mechanism enables the algorithm to effectively balance exploration and exploitation, thereby identifying an optimal subset of informative features for classification.

Let the position of a wolf be represented as

$$X_i = [x_1, x_2, \dots, x_n]$$

where

$$x_j = \begin{cases} 1, & \text{if the } j^{th} \text{ feature is selected} \\ 0, & \text{otherwise} \end{cases}$$

The quality of each candidate solution is evaluated using the classification accuracy obtained from the selected feature subset.

$$Fitness(X_i) = Accuracy(X_{selected}, Y)$$

where $X_{selected}$ denotes the selected features and Y represents the corresponding class labels.

3.3.1 Mathematical Formulation

The hunting process of GWO consists of encircling, tracking, and attacking the prey. During optimization, the prey represents the optimal feature subset, while the wolves iteratively update their positions toward the best solution. The distance between a wolf and the prey is calculated as

$$D = | C \cdot X_p - X |$$

where X_p is the prey position, X is the current wolf position, and C is a coefficient vector. The wolf position is then updated using

$$X(t+1) = X_p - A \cdot D$$

where the coefficient vectors are defined as

$$A = 2ar_1 - a$$
$$C = 2r_2$$

and

$$a = 2 - \frac{2t}{T}$$

where t denotes the current iteration, T is the maximum number of iterations, and r_1 and r_2 are random vectors in the interval $[0, 1]$. The parameter a decreases linearly from 2 to 0, allowing the algorithm to transition smoothly from global exploration to local exploitation.

Instead of following a single leader, GWO updates each wolf by considering the three best solutions (α, β, and δ). The corresponding distances are computed as

$$D_\alpha = | C_1 X_\alpha - X |,$$
$$D_\beta = | C_2 X_\beta - X |,$$
$$D_\delta = | C_3 X_\delta - X |.$$

The candidate positions are then determined by

$$X_1 = X_\alpha - A_1 D_\alpha,$$
$$X_2 = X_\beta - A_2 D_\beta,$$
$$X_3 = X_\delta - A_3 D_\delta.$$

Finally, the updated wolf position is obtained by averaging the three candidate positions,

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3}.$$

This position updating mechanism enables the algorithm to efficiently search the solution space and identify an optimal subset of discriminative features for bot detection.

3.4 Mutation Information

Mutual Information (MI) is a filter-based feature selection technique that measures the statistical dependency between an input feature and the target class. Unlike wrapper methods, MI evaluates each feature independently without relying on a specific machine learning classifier, making it computationally efficient and suitable for ranking informative features. In the proposed framework, MI is applied after the Grey Wolf Optimizer (GWO) to refine the selected feature subset by retaining the features that contribute the most information for bot classification.

Let X denote an input feature and Y denote the target class. The entropy of X, which represents the uncertainty associated with the feature, is defined as

$$H(X) = -\sum_{x \in X} P(x) \log P(x)$$

Similarly, the conditional entropy of X given Y is expressed as

$$H(X \mid Y) = -\sum_{y \in Y} P(y) \sum_{x \in X} P(x \mid y) \log P(x \mid y)$$

The Mutual Information between the feature and the target class is calculated as

$$MI(X,Y) = H(X) - H(X \mid Y)$$

or equivalently,

$$MI(X,Y) = \sum_{x \in X} \sum_{y \in Y} P(x,y) \log\left(\frac{P(x,y)}{P(x)P(y)}\right)$$

A larger MI value indicates that the corresponding feature shares more information with the target class and is therefore more relevant for classification.

After the Grey Wolf Optimizer identified an optimal subset of candidate features, Mutual Information was employed to rank these features according to their relevance to the target class. Rather than using a fixed number of selected features, the ranked features were evaluated using multiple subset sizes (**Top-4, Top-6, Top-8, Top-10, Top-12, Top-15, Top-18, and Top-20**) to determine the optimal trade-off between feature dimensionality and classification performance. This hybrid strategy combines the global search capability of GWO with the computational efficiency of MI, resulting in a compact and informative feature subset for subsequent machine learning classification.

4. Experimental Work

4.1 Experimental Setup

All experiments were conducted using **Python 3.12** in the **Google Colaboratory (Colab)** environment. The implementation employed **NumPy**, **Pandas**, **Scikit-learn**, and **XGBoost** libraries for data preprocessing, feature selection, model training, and evaluation. The wrapper-based feature selection algorithms, including **Genetic Algorithm (GA)**, **Particle Swarm Optimization (PSO)**, and **Grey Wolf Optimizer (GWO)**, were implemented from scratch.

The experiments were performed on the **Twitter Bot Detection dataset**, comprising **8,386 samples** described by **68 features**. After preprocessing, the dataset was divided into **80% training (6,708 samples)** and **20% testing (1,678 samples)** using stratified sampling. A **Random Forest classifier** with **3-fold cross-validation** was employed as the fitness function during feature selection. The optimization algorithms were executed using a population size of **20** and **30 iterations**, followed by **Mutual Information (MI)** for feature ranking. The selected feature subsets were evaluated using **Random Forest (RF)**, **Decision Tree (DT)**, **Logistic Regression (LR)**, **Support Vector Machine (SVM)**, and **XGBoost**. Model performance was assessed using **Accuracy, Precision, Recall, and F1-score**.

Table 4.1: Experimental Configuration

Parameter	Value
Programming Language	Python 3.12
Development Platform	Google Colab
Dataset	Twitter Bot Detection Dataset
Total Samples	8,386
Features	68
Train-Test Split	80:20
Training Samples	6,708
Testing Samples	1,678
GA Population	20 *(or the value used in your final experiments)*
PSO Particles	20
GWO Wolves	20
Maximum Iterations	30
Fitness Function	Random Forest Accuracy
Cross Validation	3-Fold
Evaluation Metrics	Accuracy, Precision, Recall, F1-score

4.2 Performance of Metaheuristic Feature Selection Methods

To evaluate the effectiveness of different wrapper-based feature selection techniques, **Genetic Algorithm (GA)**, **Particle Swarm Optimization (PSO)**, and **Grey Wolf Optimizer (GWO)** were implemented under identical experimental settings. Each algorithm was applied to the **68-feature** Twitter Bot Detection dataset using a **Random Forest classifier**with **3-fold cross-validation** as the fitness function. The objective was to maximize classification performance while reducing the number of selected features.

The comparative results are presented in **Table4.2**. GA selected **36 features** with a best fitness of **99.46%**, whereas PSO selected **33 features** with a fitness of **99.51%**. GWO achieved the highest fitness (**99.52%**) while selecting only **25 features**, corresponding to a **63.24% feature reduction**. Owing to its superior optimization performance and compact feature subset, GWO was selected as the wrapper-based feature selection algorithm for the proposed **GWO–MI** framework as shown in figure 4.1.

Table 4.2: Performance Comparison of Metaheuristic Feature Selection Algorithms

Algorithm	Original Features	Selected Features	Feature Reduction (%)	Best Fitness (%)
Genetic Algorithm (GA)	68	36	47.06	99.46
Particle Swarm Optimization (PSO)	68	33	51.47	99.51
Grey Wolf Optimizer (GWO)	68	25	63.24	**99.52**

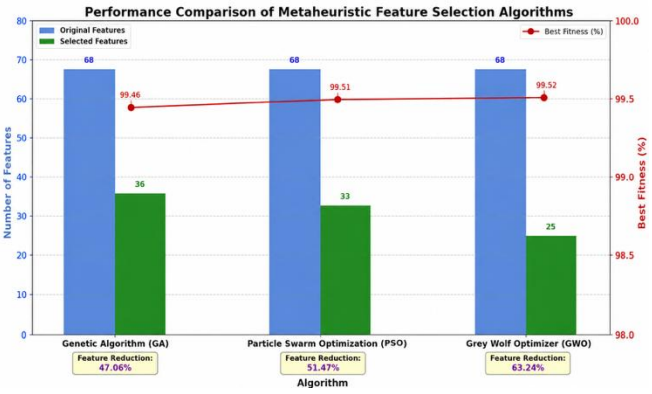


Figure 4.1. Comparative performance of GA, PSO, and GWO for feature selection, showing the original and selected features, feature reduction percentage, and best fitness values.

4.3 Performance of the Proposed GWO–MI Framework

The proposed feature selection framework integrates the **Grey Wolf Optimizer (GWO)** with **Mutual Information (MI)** to improve the efficiency of feature selection for social media bot detection. Initially, GWO was employed as a wrapper-based optimization algorithm to search for an optimal subset of informative features from the original **68-feature** dataset. Using Random Forest classification accuracy as the fitness function, GWO reduced the feature space to **25 candidate features** while achieving a best fitness value of **99.52%**. The selected features were subsequently refined using Mutual Information, which ranked them according to their dependency on the target class. This two-stage strategy combines the global search capability of GWO with the computational efficiency of MI, resulting in a compact and highly informative feature subset for classification.

To determine the optimal number of features, the ranked features were evaluated using multiple feature subset sizes (**Top-4, Top-6, Top-8, Top-10, Top-12, Top-15, Top-18, and Top-20**). The feature importance scores obtained from the MI ranking are presented in **Table 4.3**.

Table 4.3. Top-10 Features Ranked by Mutual Information

Rank	Feature	MI Score
1	favorites_followers_ratio	0.610384
2	avg_retweets	0.608246
3	avg_mentions	0.584377
4	max_tweet_length	0.485342
5	statuses_count	0.423025
6	avg_urls_status_ratio	0.392114
7	followers_count	0.379974
8	friends_followers+friends_ratio	0.351716
9	min_hashtags	0.343973
10	avg_favorite_status_ratio	0.327680

The ranked feature subsets were then evaluated using five supervised machine learning classifiers, namely **Random Forest (RF)**, **Decision Tree (DT)**, **Logistic Regression (LR)**, **Support Vector Machine (SVM)**, and **Extreme Gradient Boosting (XGBoost)**. The classification results obtained using the **Top-10** feature subset are summarized in **Table 4.5**.

Table 4.5. Classification Performance of the Proposed GWO–MI Framework (Top-10 Features)

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Random Forest	**98.75**	99.49	98.37	98.93
Decision Tree	**98.69**	99.18	98.58	98.88
Logistic Regression	**98.03**	98.27	98.37	98.32
Support Vector Machine	**94.22**	92.19	98.47	95.23
XGBoost	**98.81**	99.69	98.27	98.98

The results demonstrate that the proposed GWO–MI framework effectively preserves the most discriminative features while significantly reducing the dimensionality of the dataset. Among the evaluated classifiers, **XGBoost** achieved the highest classification performance, while **Random Forest** and **Decision Tree** also produced consistently high accuracies. Although **Support Vector Machine** exhibited comparatively lower performance, it maintained a high recall, indicating its effectiveness in identifying bot accounts. Overall, the proposed framework successfully balances feature reduction and predictive performance, providing an efficient feature selection strategy for machine learning-based social media bot detection.

4.4 Feature Subset Optimization Analysis

To investigate the influence of feature subset size on classification performance, the features ranked by Mutual Information were evaluated using multiple subset sizes, including **Top-4, Top-6, Top-8, Top-10, Top-12, Top-15, Top-18, and Top-20**. Each feature subset was independently evaluated using five supervised machine learning classifiers, namely **Random Forest (RF), Decision Tree (DT), Logistic Regression (LR), Support Vector Machine (SVM), and XGBoost**. This analysis was performed to identify the optimal number of features that maximizes classification accuracy while minimizing feature dimensionality.

The classification results obtained for different feature subset sizes are presented in **Table 4.6**. The experimental findings indicate that the optimal number of selected features varies

depending on the classifier. Random Forest achieved its highest accuracy of **98.81%** using **12 and 18 features**, whereas Decision Tree obtained its best performance of **98.99%**with **15 features**. Logistic Regression achieved a maximum accuracy of **98.09%** using **20 features**, while XGBoost consistently produced the best overall performance, achieving **98.87%** accuracy with **8 and 12 selected features**. Although Support Vector Machine exhibited comparatively lower overall accuracy, it achieved its highest performance (**95.29%**) using only **4 features**, indicating that it benefits from a smaller and more discriminative feature subset.

Table 4.6. Classification Accuracy (%) for Different Feature Subset Sizes

Features	RF	DT	LR	SVM	XGBoost
4	98.69	98.63	97.68	**95.29**	98.69
6	98.75	98.39	97.74	93.98	98.81
8	98.69	98.57	98.03	94.22	**98.87**
10	98.75	98.69	98.03	94.22	98.81
12	**98.81**	98.69	97.79	94.22	**98.87**
15	98.75	**98.99**	98.03	92.19	98.75
18	**98.81**	98.87	98.03	92.19	98.75
20	98.75	98.81	**98.09**	92.19	98.81

4.5 Comparative Analysis with Existing Methods

The proposed **GWO–MI** framework was compared with the published **GA–MI** approach to evaluate its effectiveness. Unlike the baseline study, which employed only the Genetic Algorithm, the proposed framework investigates three metaheuristic optimization algorithms (**GA, PSO, and GWO**) under identical experimental conditions. Among them, **GWO** achieved the highest fitness (**99.52%**) while reducing the feature space from **68 to 25 features (63.24%)**. Furthermore, the proposed framework systematically evaluated multiple feature subset sizes (**Top-4 to Top-20**) and included **XGBoost** in addition to the four classifiers used in the baseline study. Although the published method reported slightly higher accuracy for some classifiers, the proposed framework provides a more comprehensive evaluation of feature selection strategies and demonstrates competitive classification performance with improved feature reduction as shown in table 4.7.

Table 4.7. Comparison between the Published GA–MI Framework and the Proposed GWO–MI Framework

Aspect	Published GA–MI	Proposed GWO–MI
Metaheuristic Algorithms	GA	**GA, PSO, GWO**
Best Wrapper Algorithm	GA	**GWO**
Original Features	68	68
Selected Features	36	**25**
Feature Reduction	47.06%	**63.24%**
Feature Ranking	MI	MI
Feature Subset Evaluation	Fixed	**Top-4 to Top-20**
Classifiers	RF, DT, LR, SVM	**RF, DT, LR, SVM, XGBoost**
Best Accuracy	**99.22% (DT)**	**98.87% (XGBoost)**

5. Conclusion

This study proposed a **Grey Wolf Optimizer with Mutual Information (GWO–MI)** based feature selection framework for social media bot detection. The proposed approach effectively reduced the original **68-feature** dataset to **25 informative features** using GWO, followed by MI-based feature ranking. Experimental results showed that **GWO**achieved the highest optimization fitness (**99.52%**) among the evaluated metaheuristic algorithms (GA, PSO, and GWO). Furthermore, the proposed framework systematically evaluated multiple feature subset sizes, with **XGBoost** achieving the best classification accuracy of **98.87%**. Overall, the proposed GWO–MI framework demonstrates effective feature reduction while maintaining competitive classification performance. Future work will focus on integrating deep learning models, graph-based features, and larger real-world social media datasets to further improve bot detection performance.

Funding and Support

The authors declare that no funds, grants, or other financial support were received for the preparation of this manuscript.

Data Availability

The dataset used in this study is the publicly available **Dataset for Supervised Bot Detection on Twitter (1.0)**, which contains the Twitter account-level features used for the development and evaluation of the proposed feature selection and bot detection framework. The complete dataset is openly accessible through the **Zenodo repository** under record **5574403**. Researchers can access and reuse the dataset from the following repository: Zenodo Dataset – Record 5574403.

REFERENCES

[1]. Aljabri, Malak, Rachid Zagrouba, Afrah Shaahid, Fatima Alnasser, Asalah Saleh, and Dorieh M. Alomari. "Machine learning-based social media bot detection: a comprehensive literature review." *Social Network Analysis and Mining* 13, no. 1 (2023): 20.

[2]. Abou Daya, Abbas, Mohammad A. Salahuddin, Noura Limam, and Raouf Boutaba. "A graph-based machine learning approach for bot detection." In *2019 IFIP/IEEE symposium on integrated network and service management (IM)*, pp. 144-152. IEEE, 2019.

[3]. Hayawi, Kadhim, Susmita Saha, Mohammad Mehedy Masud, Sujith Samuel Mathew, and Mohammed Kaosar. "Social media bot detection with deep learning methods: a systematic review." *Neural Computing and Applications* 35, no. 12 (2023): 8903-8918.

[4]. Efthimion, Phillip George, Scott Payne, and Nicholas Proferes. "Supervised machine learning bot detection techniques to identify social twitter bots." *SMU Data Science Review* 1, no. 2 (2018): 5.

[5]. X. Wang et al., "Input Feature Selection Method Based on Feature Set Equivalence and Mutual Information Gain Maximization", IEEE Access, vol. 7, pp. 151525–151538, 2019. https://doi.org/10.1109/ACCESS.2019.2948095

[6]. K. Yang et al., "Arming the Public with Artificial Intelligence to Counter Social Bots", Human Behavior and Emerging Technologies, vol. 1, pp. 48–61, 2019. https://doi.org/10.1002/hbe2.115

[7]. E. Alothali, K. Hayawi, and H. Alashwal, "SEBD: A Stream Evolving Bot Detection Framework with Application of PAC Learning Approach to Maintain Accuracy and Confidence Levels", Applied Sciences, vol. 13, art. no. 4443, 2023. https://doi.org/10.3390/app13074443

[8]. E. Alothali, M. Salih, K. Hayawi, and H. Alashwal, "Bot-MGAT: A Transfer Learning Model Based on a Multi-view Graph Attention Network to Detect Social Bots", Applied Sciences, vol. 12, art. no. 8117, 2022. https://doi.org/10.3390/app12168117

[9]. S. Ye et al., "HOFA: Twitter Bot Detection with Homophily-oriented Augmentation and Frequency Adaptive Attention", arXiv, 2023. https://arxiv.org/abs/2306.12870

[10]. M. Heidari, J.H. Jones Jr., and O. Uzuner, "Online User Profiling to Detect Social Bots on Twitter", arXiv, 2022. https://arxiv.org/abs/2203.05966

[11]. Dimitriadis, K. Georgiou, and A. Vakali, "Social Botomics: A Systematic Ensemble ML Approach for Explainable and Multi-class Bot Detection", Applied Sciences, vol. 11, art. no. 9857, 2021. https://doi.org/10.3390/app11219857

[12]. D. Martin-Gutierrez et al., "A Deep Learning Approach for Robust Detection of Bots in Twitter Using Transformers", IEEE Access, vol. 9, pp. 54591–54601, 2021. https://doi.org/10.1109/ACCESS.2021.3068659

[13]. S.S. Sengar, S. Kumar, P. Raina, and M. Mahaliyan, "Bot Detection in Social Networks Based on Multilayered Deep Learning Approach", Sensors and Transducers, vol. 244, pp. 37–43, 2020.

[14]. H. Khalil, M.U. Khan, and M. Ali, "Feature Selection for Unsupervised Bot Detection", Proceedings of the 2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), Sukkur, Pakistan, 2020. https://doi.org/10.1109/iCoMET48670.2020.9074131

[15]. Mbona and J.H.P. Eloff, "Classifying Social Media Bots as Malicious or Benign Using Semi-supervised Machine Learning", Journal of Cybersecurity, vol. 9, art. no. tyac015, 2023. https://doi.org/10.1093/cybsec/tyac015

[16]. A.A. 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, pp. 15–29, 2020. https://doi.org/10.1109/TNSM.2020.2972405

[17]. A.A. Daya, M.A. Salahuddin, N. Limam, and R. Boutaba, "A Graph-based Machine Learning Approach for Bot Detection", arXiv, 2019. https://doi.org/10.48550/arXiv.1902.08538

[18]. P. Dhal and C. Azad, "A Comprehensive Survey on Feature Selection in the Various Fields of Machine Learning", Applied Intelligence, vol. 52, pp. 4543–4581, 2022. https://doi.org/10.1007/s10489-021-02550-y

[19]. Y. Li, T. Li, and H. Liu, "Recent Advances in Feature Selection and its Applications", Knowledge and Information Systems, vol. 53, pp. 551–577, 2017. https://doi.org/10.1007/s10115-017-1059-8

[20]. J.V.F. Abreu, C.G. Ralha, and J.J.C. Gondim, "Twitter Bot Detection with Reduced Feature Set", Proceedings of the 2020 IEEE International Conference on Intelligence and Security Informatics (ISI), Arlington, USA, 2020. https://doi.org/10.1109/ISI49825.2020.9280525

[21]. E. Alothali, K. Hayawi, and H. Alashwal, "Hybrid Feature Selection Approach to Identify Optimal Features of Profile Metadata to Detect Social Bots in Twitter", Social Network Analysis and Mining, vol. 11, art. no. 84, 2021. https://doi.org/10.1007/s13278-021-00786-4

[22]. C. Cea, "Dataset for Supervised Bot Detection on Twitter (1.0)", Zenodo, 2021. https://doi.org/10.5281/zenodo.5574403

[23]. S. Katoch, S.S. Chauhan, and V. Kumar, "A Review on Genetic Algorithm: Past, Present, and Future", Multimedia Tools and Applications, vol. 80, pp. 8091–8126, 2021. https://doi.org/10.1007/s11042-020-10139-6

[24]. M.A. Remmide, F. Boumahdi, and N. Boustia, "Toward a Hybrid Approach Combining Deep Learning and Case-based Reasoning for Phishing Email Detection", International Journal on Artificial Intelligence Tools, vol. 33, art. no. 2450015, 2024. https://doi.org/10.1142/S0218213024500155

[25]. M.A. Remmide, F. Boumahdi, B. Ilhem, and N. Boustia, "A Privacy-preserving Approach for Detecting Smishing Attacks Using Federated Deep Learning", International Journal of Information Technology, vol. 17, pp. 547–553, 2025. https://doi.org/10.1007/s41870-024-02144-x.

[26]. Araujo, Alex Medeiros, Anderson Bergamini de Neira, and Michele Nogueira. "Autonomous machine learning for early bot detection in the internet of things." *Digital Communications and Networks* 9, no. 6 (2023): 1301-1309.

[27]. Velasco-Mata, Javier, Víctor González-Castro, Eduardo Fidalgo, and Enrique Alegre. "Real-time botnet detection on large network bandwidths using machine learning." *Scientific Reports* 13, no. 1 (2023): 4282

[28]. Narayan, Nirdhum. "Twitter bot detection using machine learning algorithms." In *2021 fourth international conference on electrical, computer and communication technologies (ICECCT)*, pp. 1-4. IEEE, 2021.

[29]. Shinan, Khlood, Khalid Alsubhi, and M. Usman Ashraf. "Botsward: Centrality measures for graph-based bot detection using machine learning." *Computers, Materials and Continua*74, no. 1 (2022): 693-714.

[30]. Taher, Fatma, Mahmoud Abdel-Salam, Mohamed Elhoseny, and Ibrahim M. El-Hasnony. "Reliable machine learning model for IIoT botnet detection." *Ieee access* 11 (2023): 49319-49336.

[31]. A. Guendouz, F. Boumahdi, M. A. . Remmide, A. Foura, and A. Madani, “Hybrid Feature Selection Framework for Machine Learning-based Bot Detection on Social Media”, JTIT, vol. 104, no. 2, pp. 40–47, Jun. 2026, doi: 10.26636/jtit.2026.2.2541.