A Lightweight, Explainable Spam Detection System with Rüppell's Fox Optimizer for the Social Network X

Haidar ALZEYADI , Rıdvan SERT , Fecir Duran *

Posted Date: 4 September 2025

doi: 10.20944/preprints202509.0410.v1

Keywords: Ensemble learning; Spam detection; Ruppell's Fox Optimizer optimization algorithm; Explainable Artificial Intelligence

Article

A Lightweight, Explainable Spam Detection System with Rüppell's Fox Optimizer for the Social Network X

Haidar AL Zeyadi [1], Rıdvan Sert [2] and Fecir Duran [2,*]

[1] Computer Science Department, Graduate School of Informatics, Gazi University, Ankara, Türkiye

[2] Computer Engineering Department Engineering Faculty of Technology, Gazi University Ankara, Türkiye

* Correspondence: fduran@gazi.edu.tr

Highlights

- This study pioneers the adoption of Rüppell's Fox Optimizer (RFO) in Cybersecurity domain and decrease load of system.
- The RFO algorithm is employed for building a lightweight interpretable spam account detection model for X.com.
- Explainable Artificial Intelligence (XAI) via Swarm and summary Shapley values chart utilized to explaining predictions in spam detection model.
- The proposed approach achieves superior results with real world X.com datasets.

Abstract

Effective spam detection system is an essential in online social media networks (OSNs) and cybersecurity, that directly influencing the quality of decision-making pertaining to security. Conventional machine learning (ML) methods, including Decision Trees (DT), K-Nearest Neighbors (KNN) and Logistic Regression (LR), were utilized in tackling this issue. Despite their effectiveness, such methods frequently encounter known as "black box" problem, an interpretability deficiency that constrains its deployment into security applications, which comprehending the rationale of classification processes is crucial for efficient threat evaluation and response strategies. To overcome this limitation, this study employs concepts of Explainable Artificial Intelligence (XAI) to propose a interpretable model for X spam account detection. This work introduces an innovative spam detection system utilizing ensemble learning AdaBoost (Adaptive Boosting) method augmented by a carefully crafted framework. The approach employs clean data with data preprocessing, feature selection using a swarm-based, nature-inspired meta-heuristic Ruppell's Fox Optimizer (RFO) Optimization Algorithm which for the first time applied to Cybersecurity and for interpret model prediction Shapley values are computed and illustrated through swarm and summary chart. The model attained a notably accuracy of 99.35% along with high precision, recall and F1-score, surpassing conventional algorithms including DT, KNN and LR in all performance metrics. The results validate the efficacy of the suggested approach, providing an accurate and understandable model for spam accounts identification. This research represents a notable progress in the domain, offering a thorough and dependable resolution for spam accounts detection issue.

Keywords: ensemble learning; spam detection; ruppell's fox optimizer optimization algorithm; explainable artificial intelligence

1. Introduction

 doi:10.20944/preprints202509.0410.v1

In the modern era of humanity, Online Social Networks (ONSs) have emerged as the main information source X, previously referred to “Twitter”, is currently one of the favorite prominent and extensively utilized OSNs platforms. Consequently, it contributes significantly to online discussions and links millions of users .Nonetheless, their significant impact rendered it as appealing target for nefarious individuals aiming to control and sway public perspectives and decision processes [1,2]. X was frequently the main goal because of its free structure as well as growing users demographic. X platform, considers spam an essential issue and employs multiple filters for spam to safeguard users. [3–8]. X spam accounts created a new techniques and behaviors. Consequently, the utilization of new adaptive and robust technologies for detecting spam X accounts has become imperative. The use of artificial intelligence (AI) with machine learning (ML) present a viable solution. Models using ML can acquire learn from real world datasets by employing algorithms such as Ensemble Learning, DT, KNN and LR to differentiate successfully among spam and non-spam account. Such techniques provide a more powerful and dynamic methods by being able to adjust to modern spamming strategies [9,10].

A major challenge in implementing ML models refers to the "black box" , ML are black-box models and comprehending how they predict is a challenging endeavor, especially in cybersecurity activities such as spam detection, where the predictions from these models are intricate to interpret cybersecurity specialists to comprehend. Consequently, architecture must be developed to clarify the predictions of ML spam detection systems. The resolution for some such problems has manifested as the XAI approach [11]. This approach seeks to address this gap of elucidating predictions generated by ML models within the realm of cybersecurity applications [12] . XAI clarifies the black box system through Interpreting its operations and predictions.

This study highlights the outlined needs through offering an innovative approach, for develop an ML- driven spam detection system that is light-weight, where the trained model using the optimal subset of features. This facilitates a reduction in computation resources for spam detection .This work provides for feature selection method which utilized a swarm-based nature-inspired meta-heuristic method RFO [13] . Moreover, the proposed system is both learned and evaluated via ML methods. The predicted outcome generated via a learned ML-driven spam detection model is elucidated through employing the SHapley Additive exPlanations' (SHAP) Shapley values [14] .

This study primary contributions are delineated as follows:

1. An innovative XAI-powered Ensemble learning model significantly improving classification accuracy X spam account detection.
2. A swarm-based, nature-inspired meta-heuristic method, called Ruppell's Fox Optimizer (RFO) algorithm for feature selection which for the first time applied to cybersecurity and light weighted system load.
3. The proposed solution is experimentally evaluated using the real world X dataset. Developed model significantly improving performance metrics such as confusion matrix precision, recall, accuracy, F1-score and (AUC) value of area under the curve.
4. The prediction made by ML- driven spam detection model is interpreted via computing the Shapley values through the SHAP methodology.

The remainder of this work is structured as follows: In the second section, Background about ensemble Model (AdaBoost), XAI and ML algorithms. The third section presents a summary of experiments employing artificial intelligence methodologies, including ML, DL, and FL, for X social network spam detection, along with their outcomes. The fourth section clarifies the methods and materials are employed for X social network spam identification, as well as the performance metrics utilized to assess the efficacy of the applied methodologies. The fifth section presents the outcomes of the experiments for all models and discusses their findings. The final section presents overarching assessments on the outcomes.

2. Background

 doi:10.20944/preprints202509.0410.v1

The next section provides a detailed overview of the key concepts and classification algorithms underlying of proposed spam detection system. Furthermore, the explainable artificial intelligence (XAI) , SHAPE methods which employed to interpret the model's decision-making processes.

2.1. Ensemble Model

Ensemble learning in the ML field seeks to improve overall performance and generalizability by integrating the predictions of various classifiers or regression models. In this approach, aggregates various weak learners to counterbalance their individual faults, resulting in a more solid composite model. Three principal ensemble learning strategies are defined as bagging, boosting and stacking; model and error diversity play a key role in these strategies. Thus, ensemble approaches alleviate overfitting inclinations and the accuracy deficiencies seen in individual models, attaining enhanced performance in tasks necessitating high precision, such as social networks spam identification [15,16] .

This study employs the boosting algorithm AdaBoost. AdaBoost is an iterative improvement technique wherein weak learners are taught consecutively, with increased focus on previously misclassified instances at each stage. AdaBoost constructs its final decision through weighted voting for each weak learners, utilizing decision stumps as base learners. This system allows basic classifiers to collaboratively attain elevated accuracy. Moreover, AdaBoost's nonparametric characteristics and its inherent technique for mitigating overfitting enable its application across multiple challenge areas. [17,18] .

2.2. Explainable Artificial Intelligence

The imperative for understanding the procedures for making decisions of AI models has led to the emergence of XAI. The principal aim of this approach is to transparently, transparently reveal what, why and how a model predicts, the rationale behind those predictions[19], and the methodology employed in reaching its conclusions as in Figure 1.

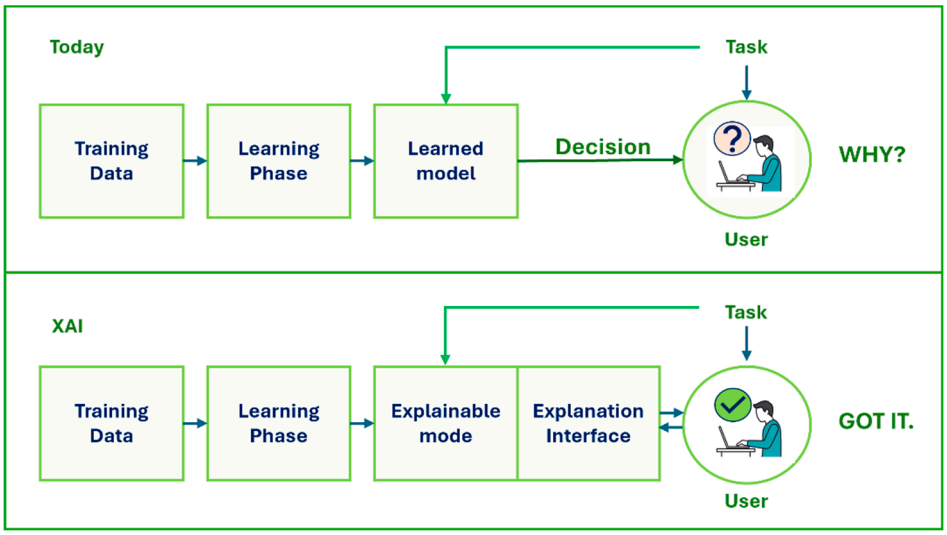


Figure 1. Comparison of traditional learning model and Explainable artificial intelligence model.

Making the internal mechanisms of black-box models understandable is critical in terms of reliability, ethical compliance and debugging. Especially in highly interactive applications such as social networks platforms spam detection, the explainability of the decisions taken as in Figure 2. is vital for both user satisfaction and compliance with legal regulations [20] .

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 September 2025 doi:10.20944/preprints202509.0410.v1

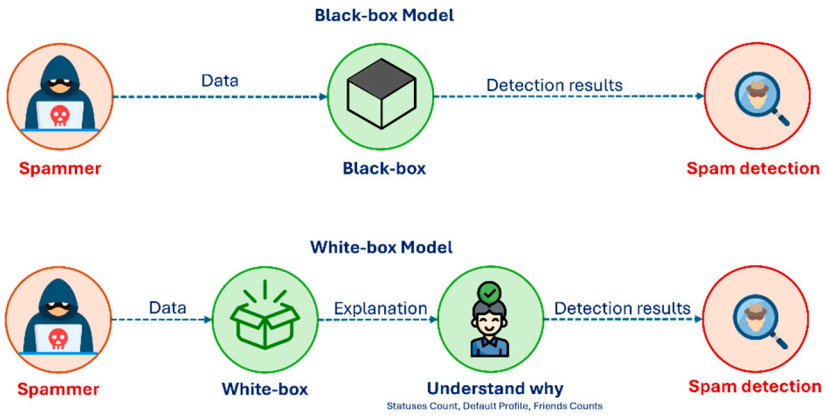


Figure 2. Explainable AI in Social Network Security: AI-Powered Spam Detection.

SHAP (SHapley Additive Explanations) produces model explanations at both global (entire dataset) and local (individual instance) levels using Shapley values derived from cooperative game theory. as in Figure 3. The contribution of each feature to the decision process is calculated by averaging the marginal effects over all feature combinations [21,22]

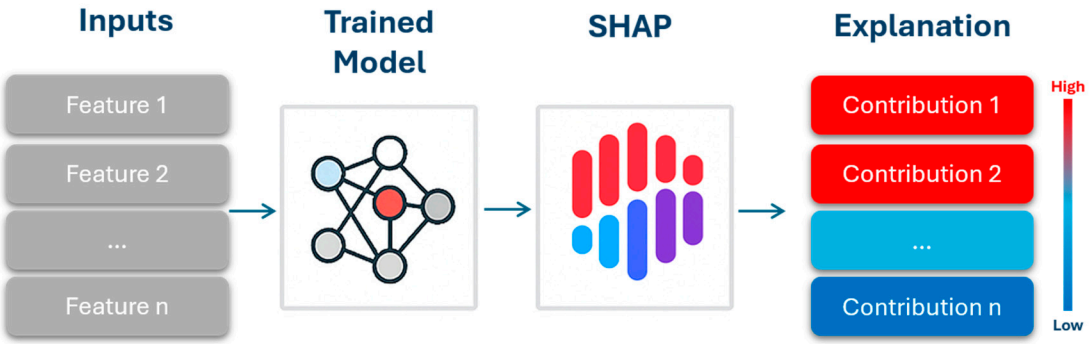


Figure 3. The operational mechanism of SHAP.

2.3. Machine Learning Algorithms

The comparison analysis part is a crucial element of this research, providing an overall structure for evaluating the effectiveness of the proposed AdaBoost Classifier. To this end, various established techniques were chosen for comparison, including DT, KNN and LR.

2.3.1. Decision Tree

A decision tree is a supervised machine-learning algorithm widely used in classification and regression problems. The model divides the data into subsets in successive internal nodes starting from the root node and the final class or numerical output is produced in the leaf nodes [23] . Data splitting decisions are usually based on measures such as Entropy Information Gain or Gini Index. The aim is to reduce the irregularity of the target variable as much as possible in each split [23]. Decision trees provide understandable models on their own thanks to their interpretable structures; they also play a critical role as the base learner in ensemble techniques such as AdaBoost. These ensemble models increase accuracy by balancing the high variance of individual trees. In text classification applications such as spam detection in social networks, decision trees are preferred due to both their fast prediction time and explainability advantages; they provide high accuracy alone or in the ensemble [24] .

2.3.2. K-Nearest Neighbors (KNN)

K-Nearest Neighbors know as (KNN) represents a non-parametric, lazy learning technique utilized for regression and classification tasks. Instead of developing a definitive model, all

 doi:10.20944/preprints202509.0410.v1

calculations take place during the prediction phase [25] . The core idea entails designating a query instance with the label (or value) that is most prevalent over its k nearest neighbors within the learning dataset [26]. KNN's straightforward implementation and its independence from any assumptions about the underlying data distribution have rendered it a widely adopted method in numerous domains, including spam social network detection [27].

2.3.3. Logistic Regression (LR)

Logistic Regression represents a fundamental ML technique employed in classification applications. It creates a model of the probability that an input falls into a specific class. The objective of logistic regression is to develop a model capable of making a classification decision using sigmoid function [28]. The high interpretability and computationally efficient nature of Logistic regression make it stand out in both explanatory analysis and practical applications like spam detection [29]. For example, models that predict social media interactions based on graph and content features can clearly reveal the effects of variables thanks to the understandable structure of linear regression [30].

3. Literature Review

The expanding problem of spammer identification in online social networks has attracted considerable focus in both academic and industrial domains. Although conventional ML methods have proven essential in addressing this difficulty, they tend to show an imbalance among evaluation metrics and explainability [31–33] . The present overview of literature seeks for a thorough review of the latest approaches and technology for spam identification. includes an especially emphasis on the ensemble learning models and explanation AI methodologies in spam identification. Regarding ML for spamming detection, in [34] proposed an approach that used trained XGBoost model with Random Forest classifier as feature selection and explain the results using XAI methods on OSN datasets, producing outstanding outcomes. Furthermore, [35] fuzzy-inference-systems (FLSs) employing Interval-Type-2 and Type-1 was employed deploying SVM, BPM, Avr Prc and LR algorithms to demonstrate their effectiveness in detecting spamming accounts.Type-2 Mamdani FLSs showed excellent results, of 0.955,0.957, 0.967 and 0.962 as accuracy, precision, recall, F-score respectively. [36] contended that BERT and CNN as a classifier exceed of SVM, RF, and NB , While the impact of such techniques depends on the characteristics set used in identifying spamming. [37] formulated a framework using a probabilistic clustering technique for tackling classification challenges caused by hostile discourse within the X network. They categorized the acquired X posts via crowdsourced specialists into two classifications, those containing hostile discourse or not included, using the samples retrieved over the Bayesian classifier, they produced the attribute they represented using the (TF-IDF). Researchers subsequently employed FL method to classify hostile discourse based on this data. they attained outcomes of 0.9453, 0.9254 , 0.9174 ,0.9256 as accuracy, precision, recall and F-score respectively. [38] utilized random under-sampling (RUS) and random over-sampling (ROS) techniques for non-spam and spam classification. Authors conducted a comparative analysis of their generated models by implementing the ensemble learning, RUBoost, K-nearest neighbors (KNN),Bayesian classifier (NB),C4.5 decision tree models using Weka. The best precision average of ensemble learning among 82% and 90% While the true positive (TPR)rate about75%. [39] suggested a Hierarchical Meta Path Score (HMPS) approach for spamming identification utilizing their contact numbers on X for promote campaigns. with suggested model campaigns data on 3,370 was obtained from the of 670,251 X users .The findings of the study were acquired using social networks and diverse features, that were created utilizing this approach. For assessing the ability in the developed approach , KNN, SVM , (DT), LR and RF, performance metrics obtained of precision (0.95), recall (0.90), F1-score (0.93), and (AUC) area under the ROC curve (0.92).

4. Methodology

This section delineates the comprehensive approach Figure 4. applied for solve the complex problem for X spam accounts identification. This section starts with dataset's characteristics and

 doi:10.20944/preprints202509.0410.v1

training, testing subsets. The latter succeeded with a discussion of data preprocessing strategies to guarantee framework stability. Thereafter, attention is directed towards the RFO for feature selection and then classification phases with Ensemble model.

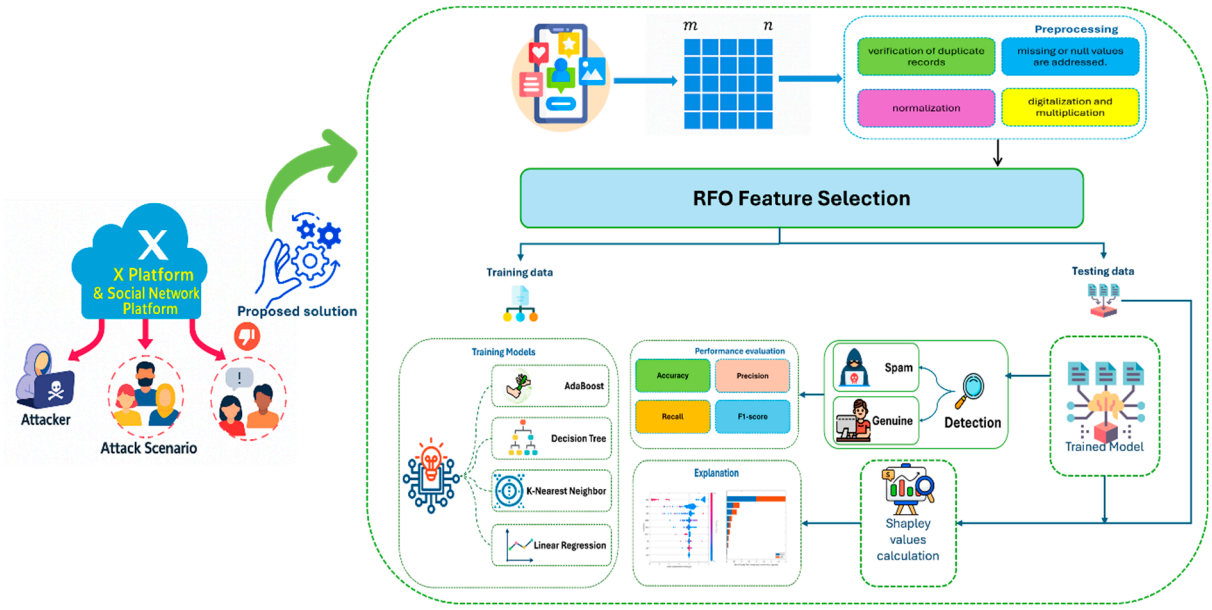


Figure 4. A proposed methodology of spam identification via social networks.

The section concludes includes a thorough assessment for model efficiency, utilizing various measures for deep evaluate the model's value completely. Ensemble learning integrates several models to provide an enhanced classifier with robust accurate results [40] .The choice for adopting the Ensemble method AdaBoost model is based on upon its shown efficacy in managing data with complex environments deemed crucial in detection X spam. The AdaBoost ensemble seems particularly for this research given the ability it has for handling the complexities of social media patterns, which is a crucial element in this detection model. The ensemble Boosted Trees method with AdaBoost, that integrates with decision trees Learner type for boost classification precision, has a unique benefit in identifying Nuanced characteristics which separate spam and authentic accounts [41,42] .

Furthermore, this work provide comparison methods that give a wider framework to assess efficacy of the model. The approach has been improved by recent strategies of model explainable that providing understanding of models make predictions mechanism.

4.1. Dataset Description

The dataset used to conduct this research was compiled by [35] . This dataset consists of 1225 each representing a X account encompasses 11 unique features, including categorical and numerical attributes, as detailed in Table 1. These attributes encapsulate account characteristics, user behavior and X metadata. This set contains 718 classified as spam accounts, while the remaining 507 are legitimate users labelled as non-spam. This establishes a dependable benchmark for the classification of spam versus non-spam. The dataset encompasses a combination of user profile attributes and X posts -level metrics that are effective in differentiating spammers from authentic users.

Table 1. The dataset's taxonomy and criteria ranges.

No	Input model Features	Type	Range of evaluate
1	User Statuses Count (USC)	Integer	0-99,100-199,…,1000000-1999999

 doi:10.20944/preprints202509.0410.v1

2	Sensitive Content Alert (SCA)	Boolean	TRUE(T)/FALSE(F)
3	User Favorites Count(UFC)	Integer	0-9,10-19,20-29,…,100000-1999999
4	User Listed Count (ULC)	Integer	0-9,10-19,20-29,…,900-999
5	Source in Twitter (SITW)	String	Yes(Y)/No(N)
6	User Friends Counts (UFRC)	Integer	0-9,10-19,20-29,…,1000-99999
7	User Followers Count (UFLC)	Integer	0-9,10-19,20-29,…,100000-1999999
8	User Location (UL)	String	Yes(Y)/No(N)
9	User Geo Enabled (UGE)	Boolean	TRUE(T)/FALSE(F)
10	User Default Profile Image (UDPI)	Boolean	TRUE(T)/FALSE(F)
11	Re-Tweet (RTWT)	Boolean	TRUE(T)/FALSE(F)
12	Account-Suspender (CLASS)	Boolean	TRUE(T)/FALSE(F)

Furthermore, features category criteria of the utilized dataset as detailed below:

- **User profile features**: Several attributes measure activity of account with its reach. User-Statuses-Count (USC) represents the user's latest Tweets or retweets. This indicates the account's productive nature on the platform. Moreover, User-Followers-Count (UFLC) is the total quantity of Tweets this user has endorsed over the account's existence. Whereas User-Friends-Count (UFRC) is users number following this account Also known as their "followings". User-Favourites-Count (UFC) perspectival denotes if the verifying user has liked (favorited) this Tweet. Additionally, User-Listed-Count (ULC) this indicates of public lists number of which the user is a member, its measure prominence of account among different users. These number indicators essentially encapsulate engagement of accounts and footprint in social platform. including these properties enables the analysis of such patterns within the dataset.
- **Tweet and profile indicators:** Dataset additionally includes a number binary (yes/no) attributes that represent tweet characteristics and profile configurations. Sensitive-Content-Alert (SCA) represents a Boolean value It denotes sensitive items contained in Tweet content or within the textual properties for the user's property. This may indicate spam, as spammer tweets often contain this type of material. Source-in-X (SITW) displays the utility employed to publish the Tweet, formatted as an HTML string. denotes whether the tweet was disseminated using an official X platform (YES) or a third-party source (NO).For instance, Tweets originating from the X website possess a source designation of web. Conversely, spammers may utilize specific applications. User-Location (UL) constitutes a category domain indicates the user provided location for the account profile (YES) or unfilled (NO). the location actual text is not employs due to its random nature and difficulty in parsing. This binary characteristic just indicates the existence of a profile location. commonly possessed by legal users. User-Geo-Enabled (UGE) if the user has activated geotagging on tweets is TRUE (indicating that the account permits the attachment of geographical coordinates to tweets). User-Default-Profile-Image (UDPI) is a boolean variable, signifies that the user using default X profile picture(When true, indicates the used not uploaded image for profile). Significantly, suspicious accounts typically use the default image profile and possess minimal private information [42] .So this trait may indicate a potential threat. Finally, ReTweet (RTWT) Boolean value, this indicates if the validating user has retweeted this Tweet.

 doi:10.20944/preprints202509.0410.v1

- **Class attribute**: this attribute in the dataset signifies that an account is categorized as spam account if true, while false represents an account that is legal.

Overall, this dataset offers a comprehensive overview of X account attributes, integrating user profile metrics with content indicators. The fundamental reason for providing an extensive overview of the data is the essential requirement for a deep comprehension of the dataset's intricacies prior to initiating ML models. There is essential for constructing a robust spam identification system.

4.2. Data Preprocessing and Balancing

This section discusses the preprocessing conducted on the X spam dataset for preparing to proposed model. The objective consists of cleanse data, address any anomalies, and convert it into a structure that fits ML techniques. The procedure started with the verification of duplicate records. that could occur as a result of challenges faced via collection data. Subsequent to this step, every single row constitutes a distinct data element. Subsequently, missing or null values are addressed. Although such elements are infrequent, they are excluded to prevent bias during the subsequent scaling process. Following stage data digitalization and multiplication. In this stage process of encode all binary features values with “Yes-No” or Boolean “True-False” into numerical values "1-0". additionally, given data that is labelled by the interval criteria, for each data generated five different random real values in the confidence interval for the pertinent criterion. Here, feature with binary values in corresponding data to the confidence interval criterion also multiplication by five like identical value. Data comprising 1,225 parts was transformed into 6,125 parts of data upon completion of this process. The final result from these preprocessing techniques is a refined dataset suitable for ML models. It maintains with identical of rows quantity (excluding duplicates that were eliminated) with same features set, but use a uniform numeric format.

Every row represents the 11 input unique features, five normalized continuous elements(indicating the user's status count, follower count, etc.) and six binary indicators (for indicators such as sensitive content, default picture, etc) and finaly binary class label. The clean dataset split into two subsets with 80:20 ratio for model training and testing set. This partition is essential during training and validation stages of models. The training subset is applied to construct the model. Whereas, the testing subset is designated for assessing the model's performance on previously unobserved data. The test ratio is determined based on research results that indicates this percentage offers an optimal equilibrium throughout training and testing [43]. The selection is further affected with the requirement of an adequately sized testing subset for validate assess the model's performance. Through the execution of these preprocessing activities ensure that detection model are able to make efficient use of the information without being hindered during the process. This careful preprocessing steps provide a solid foundation for building an accurate spam detection model.

The complete dataset of 6125 samples was divided by a 80% for training and 20% testing stratified split, producing a training set of 4900 samples (2872 spam and 2028 non-spam) and a test set of 1225 samples (718 spam and 507 non-spam). In ML, class imbalance can make predictive models favor the majority class. consequently, compromising performance on underrepresented outcomes. To tackle this challenge, the current study employs the Synthetic Minority Over-sampling Technique (SMOTE) to achieving a more equitable class distribution while the development of models. SMOTE was employed for the training set to expand the non-spam category samples from 2028 to 2872. Creating a meticulously balanced training set of 5744 samples (2872 spam and 2872 non-spam) ensures balanced class distribution for the model training as shown in Figure 5.

 doi:10.20944/preprints202509.0410.v1

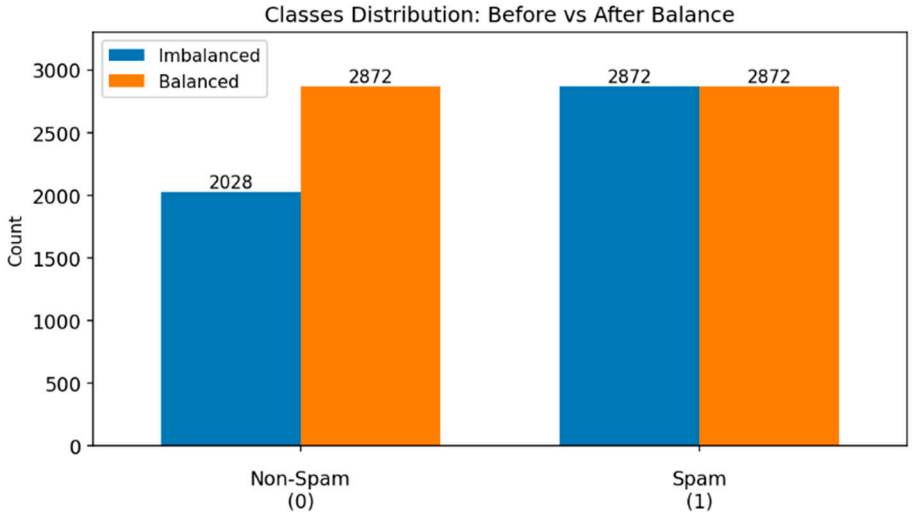


Figure 5. Classes distribution before vs after SMOTE.

4.4. Feature Selection Approach

Feature selection is an essential process within ML which significantly impacts model efficacy. Not every feature within dataset inherently aids the model's training procedure. Extraneous features might diminish the ability to classification thus impede the learning phase. Selecting best feature subset that beneficially impact the model streamlines it, facilitating the achievement of more precise and expedited outcomes.

4.4.1. Rüppell Fox Optimization

In ML domains, traditional optimization techniques frequently prove difficulties due to the expansive and intricate solution spaces encountered. Meta-heuristic methods provide efficient solutions to address this constraint. These algorithms apply random search techniques to locate solutions near the global optimum. Meta-heuristic techniques, sometimes inspired from natural events, are especially beneficial for selection feature tasks because of their low computing expense with excellent precision. This work leverages RFO, swarm-based inspired by the hunting and survival behaviors of Rüppell's foxes, for feature selection. The RFO systematically balances between local exploitation with global exploration to ascertain optimal solutions. Its start with initial with randomness-based steps to investigate various areas and amplifies its search near the most promising answers to expedite convergence. RFO creates random initial positions in each dimension after establishing the upper and lower boundaries of the solution space, as shown by the following equation(1-3), the population size, maximum iterations, and algorithm-specific parameters are established at this stage. In each iteration, a P value randomly is initially established within the interval (0, 1).

$$y_j^i = l_j + r \times (u_j - l_j) \tag{1}$$

y_j^i : *The jth dimensional coordinate of the ith member within population*

u_j : *The upper limit of the* $j-th$ *dimension*

l_j : *The lower limit of the* $j-th$ *dimension.*

r : *A random number generated within the interval* [0,1].

The daytime mode is activities if p is greater than or equal to 0.5; otherwise, the nighttime mode is activities. The Rüppell's fox's senses and the update techniques the algorithm uses throughout each

 doi:10.20944/preprints202509.0410.v1

iteration are determined by the daylight and night modes. The following equations are used to calculate the senses of eyesight, hearing, and smell:

$$s = \frac{1}{1 + e^{(K/2-k)/100}} \quad (2)$$

$$h = \frac{1}{1 + e^{(k-K/2)/100}} \quad (3)$$

$$smell = \frac{0.1}{\left|acos\left(\frac{2}{1 + e^{(K/2-k)/100}}\right)\right|} \quad (3)$$

where K is the maximum iteration and k is the present iteration. The population size, maximum numbers of iterations, and parameters of algorithm are initialized during this phase. In each iteration, a random value p is evenly selected from the interval [0,1]. If p < 0.5, the algorithm functions in night mode; otherwise, it transitions to daytime.

4.1.2. Daylight and Night Behavioral Update Strategies

At each iteration, a random limit variable rand is sampled uniformly from (0,1) to select between daylight mode (p≥0.5) and night mode (p<0.5). These modes determine which sensory-driven update rule the Rüppell's fox employs:

1. Daylight mode (p≥0.5)

1.1. If s≥h and rand≥0.25, update position using eyesight; otherwise (rand<0.25), update via eye movement.

1.2. If s<h and rand≥ 0.75, update position using hearing; otherwise (rand<0.75), update via ear movement.

2. Night mode (p<0.5)

2.1. If s<h and rand≥0.25, update position using hearing; otherwise (rand<0.25), update via ear movement.

2.2. If s≥h and rand≥0.75, update position using eyesight; otherwise (rand<0.75), update via eye movement.

In both modes, the foxes also employ their olfactory sense. As a fox approaches its prey, its olfactory acuity increases exponentially; conversely, as it moves away, olfactory sensitivity decreases exponentially.

In the optimization process, KNN algorithm was employed to evaluate classification accuracy. A neighborhood coefficient of k=5 was chosen for both the KNN classifier and the cross-validation procedure. The RFO algorithm was initialized with a population of 100 individuals and executed for 200 iterations. From the original set of 11 features in the dataset, a subset comprising nine features was obtained. The nine features ultimately selected by the RFO algorithm are presented in Table 2.

Table 2. Features selected by the RFO algorithm.

Optimal features	Selectedf eatures
Nine optimal features for detection models	SCA, UL, UDPI, RTWT, USC, UFC, ULC, UFRC, UFLC

The features listed in Table 2 were identified by the RFO algorithm as the most influential in maximizing classification performance.

4.5. Model Training

For this stage The Ensemble approach Specifically AdaBoost was chosen, due to its reliability and capability. In Boosting approach, combining a group of weak models to create a single strong model. This procedure in ML is referred as boosting. Adaptive Boosting classification technique also known as AdaBoost [44] . Every weak learners generates a Logical expression and executes a learning

 doi:10.20944/preprints202509.0410.v1

process. Decision trees, a type of machine supervised learning, enable the continuous partitioning of data based on a specified parameter to aid in the creation of predictive models using training data, decision nodes and leaves are crucial elements for clarifying the tree's structure. Upon the completion of the process, when all classifiers produced over the iterations are ultimately solicited to cast their votes regarding the preferred result for Innovative inquiry, some trees are likely to receive a greater vote over alternatives. This is the primary distinction between AdaBoost and bagging methodologies. AdaBoost has been employed via numerous researchers in investigations on spam and fraud detection [45,46]. For evaluation step applied collection of reliable metrics to assess the model's efficacy. Incorporating Accuracy, F1-score, Recall, Precision, the Receiver Operator Characteristic (ROC) with the Area under the Curve (AUC). These performance measures were selected based of its comprehensive capacity to elucidate the model's strengths and Constraints. Hence, the structure of the training model then evaluation process was thorough and precise, integrating RFO for feature selection optimization. This multifaceted strategy guarantees an in-depth assessment, therefore enhancing the dependability and integrity of the work outcome.

4.6. Model Evaluating

Machine learning model performance evaluation is an essential part inside the comprehensive model workflow. The evaluating stage includes various types of key metrics, crafted to provide an advanced overview about a model strengths and constraints, so directing future endeavors to enhance accuracy of the prediction. Regarding our study, researchers chose to apply a set of primary metrics. In order to evaluate entire model efficiency with classification techniques employed in spam detection: Accuracy, F1-score, Recall, Precision the ROC and AUC as Table 3.

Table 3. Performance metrics for evaluating the model.

metr□cs	Formula	Descr□pt□on
Accuracy	$\frac{tp + tn}{tp + fp + tn + fn}$	It is a measure which indicates the proportion of accurately identified cases relative with the total cases assessed.
F1-score	$2 * \frac{(precision * recall)}{(precision + recall)}$	This is a statistic that provides the harmonic average of recall and precision metrics.
Recall	$\frac{tp}{tp + fn}$	This indicator quantifies the proportion of non-Spam positives identified by the training model in a specific classification issue.
Prec s □on	$\frac{tp}{tp + fp}$	It is a measure that quantifies the proportion of accurately identified positive cases among all positively classified cases. The amount of accurate predictions to all correct predictions is termed precision.

 doi:10.20944/preprints202509.0410.v1

AUC	$\int TPR\, d(FPR)$	This indicator evaluates the efficacy of the training model based ROC curve , illustrating the relationship among the rate of false positives and the rate of true positives across various thresholds.
Macro-average	$(x+a)^n = \frac{1}{C}\sum_{i=1}^{c} M_i$	Unweighted average metric values of the per-class
M□cro-average	$\frac{\sum_{i=1}^{c} TP_i}{\sum_{i=1}^{c}(TP_i + FP_i)}$	Average of the designated metric computed from the combined predicted and actual values across all classes.
We□ghted average	$\frac{\sum_{i=1}^{c} w_i \,.\, M_i}{\sum_{i=1}^{c} w_i}$	Weighted average of per-class metric values, based on the frequency of occurrence for each class

where:

- True-Positive (TP): represents the number of accounts model predicted correctly as spam.
- True-Negative (TN): represents the number of accounts model predicted correctly as non-spam.
- False-Positive (FP): represents the number of non-spam accounts the model predicted as spam.
- False-Negative (FN): represents the number of spam the model predicted as non-spam.
- M_i is metric C class, w_i is number of true instances (Support) for class i

4.7. Implementation Environment

For this work, the computing resources was carefully prepared to satisfy powerful requirements. The hardware resources configuration was based on a personal computer equipped using CPU Intel (R) Core (TM) i7-14700HX-2.10 GHz with RAM memory 32 GB. For Operating System Windows 11 Pro (64-bit) guaranteeing a reliable and effective computational performance. The research employs the Python 3.11.7 programming language, augmented by numerous libraries like Pandas, NumPy for manipulation the data and efficient numerical operations. Anaconda was utilized as platform for manage environment. MATLAB academic version R2024b was selected as implementation and deployment environment. Since its offers built-in interpretative interface for mathematical and graphical functions together with many specialized toolkits and application for advanced modeling like Statistics and Machine Learning Toolbox, classification learner app with global and local interpretation plots.

5. Results and Discussion

Our findings indicate the efficacy of explainable ML models in identifying spam accounts on online social networking like X. This section will clarify the conclusions obtained through the performed experiments within methodology for study. As illustrated in Table 4, The suggested design exhibits excellent results , achieving a final accuracy rate about of 99.35%.

Table 4. Evaluations of the proposed model's performance.

Metric/class	Precision(%)	Recall(%)	F1-Score(%)	Support
Non-Spam	99.08	98.62	99.02	507
Spam	99.03	99.86	99.44	718
Overall				1225

 doi:10.20944/preprints202509.0410.v1

Accuracy	**99.35**		
Macro Avg	99.41	99.24	99.32
Micro Avg	99.34	99.34	99.34
Weighted Avg	99.35	99.34	99.35

The precision rates for Non-Spam (0) and Spam (1) classifications are 98.61% and 99.03%, respectively. The model produces relatively low false positive, as evidenced by these extremely accurate values. This is essential in spam identification help prevent the erroneous classification of real accounts as spam. The ratio of accurate predictions to the total correct predictions is termed precision. regards to spam detection, a precision score of 99.03% within Spam label indicates the system infrequently labels a Non-Spam account as spam. Equally, precision score for Non-Spam label demonstrates that the model is exceptionally dependable in accurately detecting real accounts. The system has remarkable recall scores of 98.62% for Non-Spam and 99.86% for spam. Recall evaluates capacity of the model for recognize all pertinent cases within each category. A strong recall score of 98.62% in Non-Spam instances guarantees the accurate identification of nearly all Non-Spam accounts. In contrast, recall score of 99.86% in Spam category proves the approach effectively identifies all spam accounts, establishing a formidable initial barrier towards unsolicited accounts. The F1-Score It also referred as F-measure is a harmonious metric combining Precision and Recall. taking into account False-Positives and False-Negative. For Non-Spam, the model has an F1-Score of 98.61%, while for spam, it's 99.44%. This model's results show that it does a great job at balancing Precision and Recall, this considered a highly discerning in classification problems. Support values are 507 with Non-Spam when 718 for spam. Give the count for true instances for each category within the dataset. This values have weight since they provide information for the remaining performance metrics, including F1-Score, precision, and recall. With numerous instances for each classes, the model's powerful performance metrics are even more dependable. Overall, with higher scores of every performance indicators indicate the system's efficacy, dependability, and resilience in categorizing accounts as Spam or Non-Spam. The indicators together demonstrate the system's superior performance. rendering it an exceptionally dependable instrument for X spam accounts identification.

5.1. Evaluation of Performance with the ROC Curve

A receiver operating characteristics (ROC) graph computed to evaluation model performance solidity. The calculated area under the curve (AUC) is a crucial metric that reflects the discrimination ability of each model. AUC for spam accounts demonstrated superior performance of 0.9992, 0.9920, 0.9935, ,0.9901 and for AB, DT, KNN and LR , respectively. These results indicate strong discrimination abilities. Moreover, the AUC result of RFO -AdaBoost around 0.9992 as shown in Figure 6. is nearly ideal and signifies as a system demonstrates exceptional separation. This indicated system's capability to accurately categorize hence validating the system's solidity. A strong AUC score indicates a reduced likelihood of mistakes in classification with the system. This suggests a minimal false positive rate, it's essential with accounts spam detection. A small false positive rate indicates non-spam accounts have a lower probability of erroneously categorized as spam accounts, guaranteeing the preservation non-spam accounts.

Under perfect case, area under the ROC curve will flow with graph upper left region, indicating a false positive rate with 0 while positive rate of 1. A nearer curves adheres upper left region, the test accuracy greater. The suggested design, having a value for the AUC of 0.99, very closely matches the perfect case. Briefly, the ROC curve with associated AUC at 0.99 offer solid proof to support the system's proficiency in categorizing X accounts as non-spam or spam.

 doi:10.20944/preprints202509.0410.v1

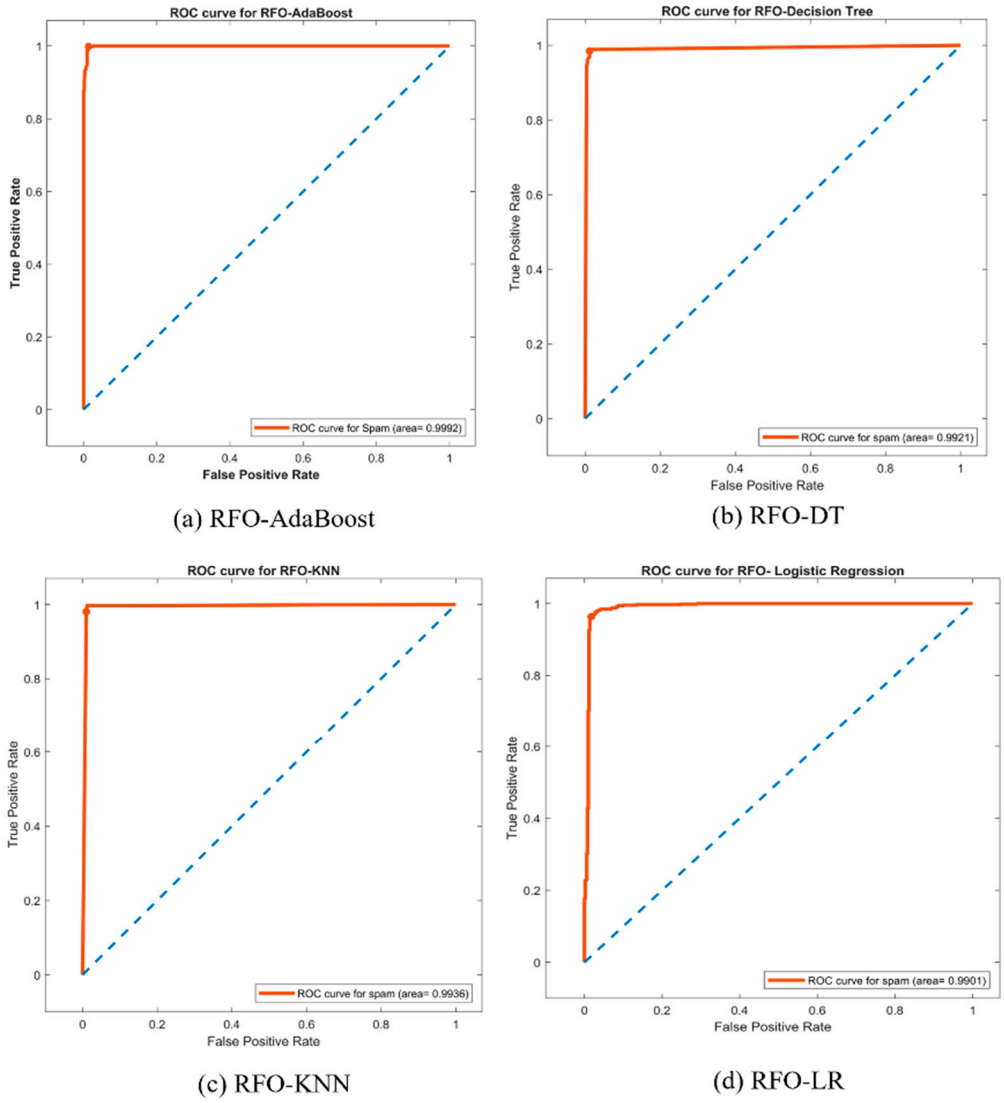


Figure 6. Depiction the ROC Curve of spam classification class utilizing nine features picked through an RFO approach.

5.2. Performance Evaluation Using Confusion Matrix

The confusion matrix is conducted as part of the experimental assessment. The model accurately in properly classifying non-spam account and detecting spam account, as well as the associated error rates, are shown in Figure 7. A comprehensive examination reveals that the non-spam accounts identification rate is roughly 99% and 96% across all detection models. Furthermore, The detection models have exhibited exceptional efficacy in precisely recognizing particular spam accounts, achieving an accuracy near or exceeding 99%.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 September 2025 doi:10.20944/preprints202509.0410.v1

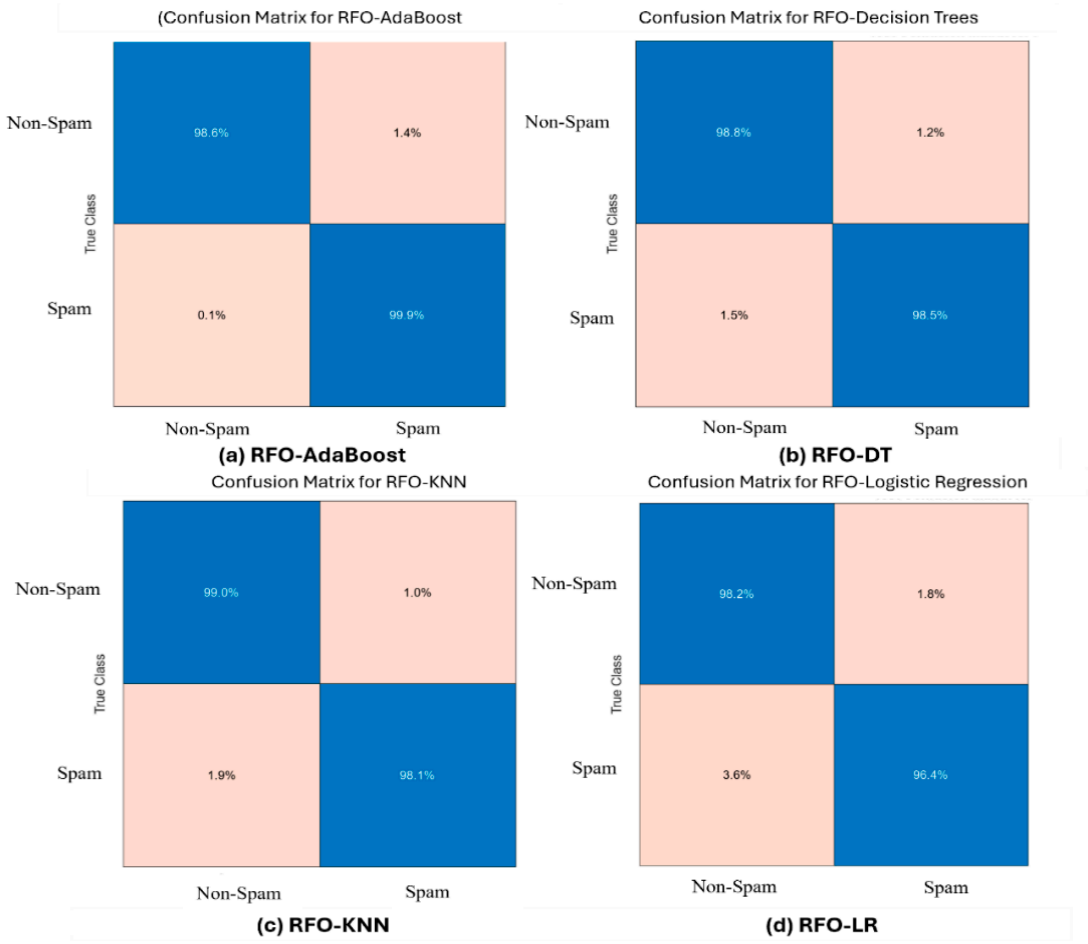


Figure 7. Visual depiction of the Confusion Matrix produced for classification utilizing the RFO technique on nine chosen features.

5.3. Explain Global Model Predictions Using Shapley Importance Plot

The SHAP analysis depicted in Figure 8. (a) functions to be a crucial technique for clarifying the intricate procedures for making decisions within the XAI-ROF- AdaBoost method employed for this research. Shapley Importance chart offers a model-agnostic based game theoretical assessment of feature impact. Such score becomes crucial towards comprehending the impact of each feature over the predictions result.

Shapley values of predictors for a collection of query points to ascertain which predictors exert the most important (or least important) average influence on the size of model output. These values elucidate the divergence of the forecast for the query point from the mean prediction, attributable to the predictor. The sign of the Shapley value denotes the direction of the deviation, while the absolute value signifies its size. This high interpretability degree is essential with models necessitating significant responsibility and transparency.

Additionally, specific features strongly influence the predicted effectiveness based on the ensemble model. Shapley Importance analysis showed SCA is the main feature affecting both spam and non-spam predictions scores. Particularly, the mean absolute Shapley values demonstrate that altering SCA modifies the expected classification score. Secondary features USC, UFC and UFRC showed a moderate impact. where each predictor spam portion partially surpasses its non-spam portion. This disparity signifies a minimally more significant influence on increasing the spam score compared to the non-spam score. UFLC, RTWT, UL, and UDPI exerted diminishing effects in succession. while ULC had the minimal influence

In conclusion, Shapley Importance highlights that content-based indicators (SCA) and high-level indicators of engagement (USC, UFC) stimulate/ reinforce spam identification. These findings

 doi:10.20944/preprints202509.0410.v1

underscore potential for focused feature engineering specifically enhancing indicators for "sensitive content" and stress the significance of XAI approaches for transparent and responsible cybersecurity frameworks.

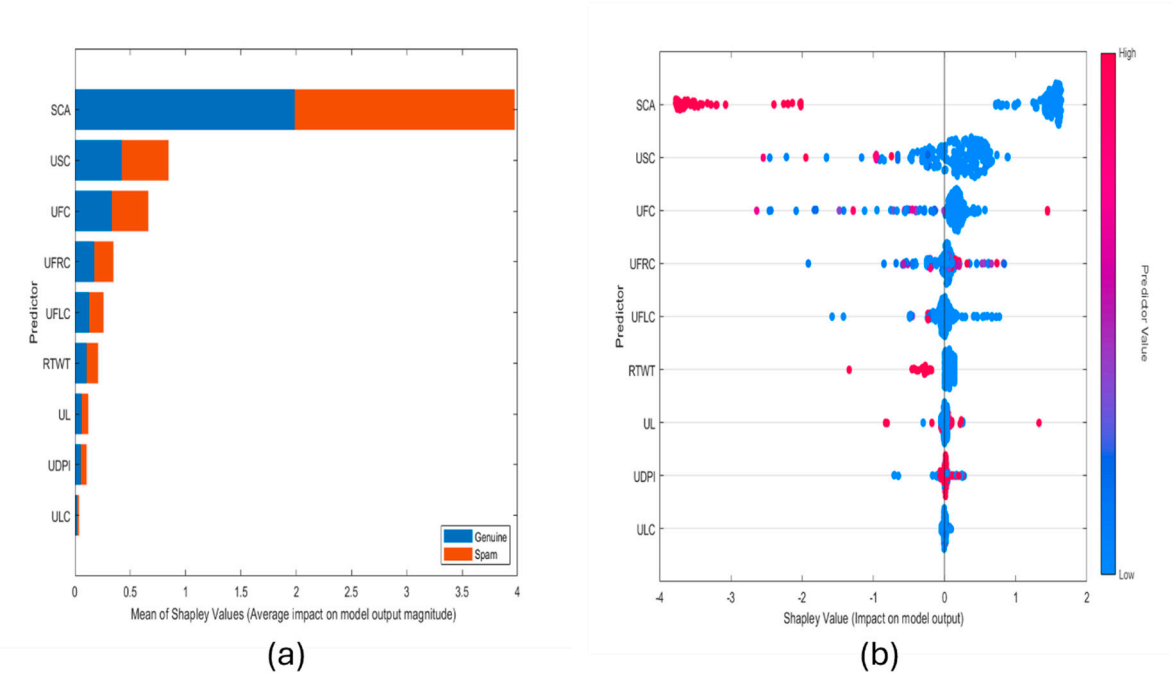
5.4. Explain Global Model Predictions Using Shapley Summary Plot

On the other hand, Shapley swarm chart Figure 8. (b) employed to comprehend the complex makes predictions processes of ML model used in this study.

Shapley swarm charts interpret the influence of individual indications on model classification. The Positive SHAP values suggest that these features predominantly influence model classification in direction spam account. Conversely, negative values move the model towards the non-spam account. This deep grasping is essential for situations where explainability and transparency are critical.

The features of "SCA", "USC" and "UFC" exhibit the greatest positive SHAP values. This indicates that these features are robust markers for categorizing an account as spam. These elevated SHAP values indicate significant influence in the model making predictions.

Low "SCA" feature in X accounts generated substantial positive SHAP values, thus elevating spam account probability. Conversely, high "SCA" was associated with negative SHAP values, significantly steering the model away from classifying as non-spam. The second most impactful feature was "USC". Accounts with a minimal number of total tweets (low USC) exhibited positive SHAP values, suggesting an increased probability of spam, while elevated "USC" yielded negative SHAP scores indicative of non-spam behavior. The "UFC" is ranked third, accounts with few favorites showed positive values, while active "favoriters" yielded negative SHAP values. Features with moderate influence included "UFRC" and "UFLC". Reduced "UFRC" or "UFLC" with positive SHAP value elevates the possibility of spam, when elevated counts inhibited it. Retweet feature "RTWT" showed a comparable trend, minimal retweet rate moderately inclined closer to spam, In contrast, elevated retweet activity significantly favored non-spam content. Both "UL" and "UDPI" showed lower effects, nonetheless, their patterns remained stable. The lack of a specified location and the utilization of a default user profile image marginally increased the possibility of spam, whereas, the valid location and customized avatar have marginally reduced it. Finally "ULC" demonstrated minimal impact since its SHAP values were concentrated around zero.



 doi:10.20944/preprints202509.0410.v1

Figure 8. (a) Summary plot illustrating the contribution of each attribute to the identification of X accounts categories via XAI-ROF-AB. (b) illustrates a swarm chart depicting the feature contributions to the detection of every class of X accounts utilizing an explainable spam detection System based on ROF-AdaBoos

Optimizing computational efficiency is essential for the real-time deployment and scalability of X spam detection models. Figure 9. comparing training times with RFO vs absence of RFO shows a significant enhancement in the effectiveness of training time for all assessed modeling with adaptation of RFO.

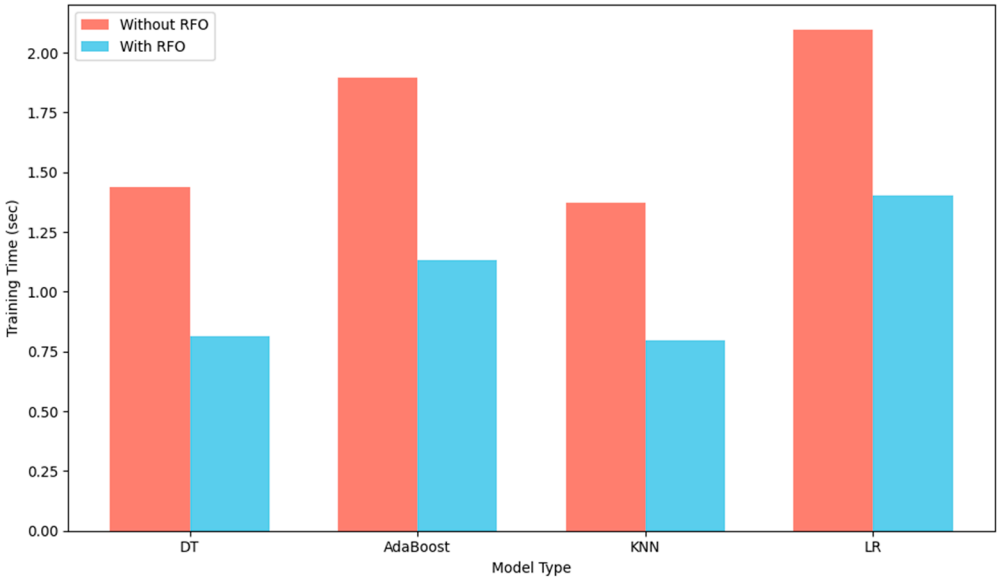


Figure 9. Impact of RFO vs absence of RFO training times Across Models.

Figure 10. depicts the influence of RFO on the prediction speed of the AdaBoost method that achieved Highest accuracy, quantified in observations per second. The chart displays two bars: performance AdaBoost baseline without RFO, the RFO improved performance. Results indicate a significant enhancement, with prediction speed rising from around 66,587 observations per second to 104,551 observations per second—a gain of over 38,000 observations per second. This boost in performance underscores the efficacy of RFO in expediting inference time, which is especially advantageous in X platform. The graphic clearly conveys the comparative efficiency and supports the incorporation of optimization strategies in ensemble models for detection spam system. Furthermore, Table 5 offers a comparative the suggested model in relation to models for conventional ML, comprising DT, KNN and LR. The comparative metrics employed are Recall, Precision, F1-Score and Accuracy. The suggested model exhibits outstanding results over every metrics attaining Recall, Precision, F1-Score and Accuracy with 99%.

 doi:10.20944/preprints202509.0410.v1

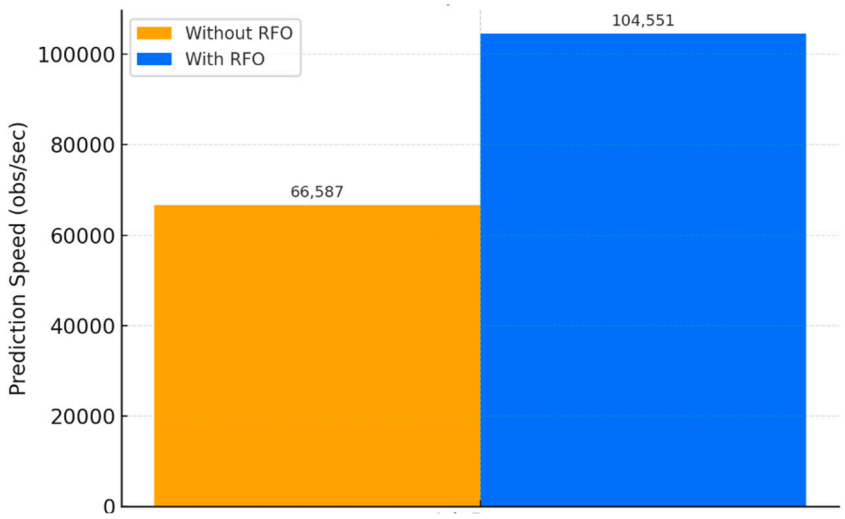


Figure 10. Impact of RFO on prediction speed of AdaBoost model.

Table 5. Comparing our model with traditional ML models.

Models	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-score (%)**
RFO-AdaBoost	99.35	99.03	99.86	99.44
RFO--DT	98.69	99.16	98.47	98.81
RFO--KNN	98.45	99.30	98.05	98.67
RFO--LR	97.14	98.72	96.38	97.47

This is especially significant considering that the subsequent highest scoring classifier DT attains 98.69% and 98.81% with Accuracy and F1-score, respectively. While shows good Recall of 98.47% and Precision of 99.16%, prove its capacity for accurately detect spam accounts with reducing the incidence both false negatives plus false positives. Model with excellent Precision guarantees that authentic accounts are not classified as spam accounts, whereas model with excellent Recall guarantees detects a significant percentage of spam accounts. Simply F1-Score constitutes the Harmonious mean between Precision with Recall, functioning to be an equitable metric to evaluate the performance of a model. The suggested model's excellent F1-Score of 99.44% signifies a successful equilibrium among Recall and Precision, this is essential with social platforms When either false negatives plus false positives carry substantial consequences.

The design suggested surpasses KNN even LR, that display results of accuracy above 98.45% and 97.14%, accordingly. Although these models provide commendable Recall or Precision results but lack performance balanced. where evidenced with the diminished F1-Scores of 98.67% and 97.47% respectively. In DT 98.47% Recall is exceptionally good. However, it has deficiencies in Precision, leading to a diminished F1-Score of 98.81%. The LR works adequately but is inferior to the suggested model, with Accuracy 97.14 and F1-Score of 97.47%.

Overall, the comparison examination highlights the effectiveness for the proposed methodology for spam accounts identification. This outstanding score over various criteria designates it to be a formidable and dependable method, particularly in contrast to conventional ML techniques. Moreover, the effectiveness and dependability of the suggested model in accounts categorization are amply supported by the comparison study in Table 5.

The elevated AUC score shown in (Figure 6.) demonstrates this outstanding results metrics, strongly advocating its implementation in social media platforms detection spam systems. Moreover, the effectiveness and dependability of the suggested model in accounts categorization are amply supported by the comparison study in Table 5.

Table 6 presents analytical comparison of related studies which employed the identical dataset, establishing a definitive standard for our method efficacy in comparison to other prominent

 doi:10.20944/preprints202509.0410.v1

techniques in the field of X spam accounts detection. That contrastive is essential for situating our method efficiency, specifically emphasizing improvements attained within our amalgamation of XAI methods along with Ensemble model, a technique rarely used with the research we compared.

Table 6. Benchmarking our research in comparison with current studies.

Author	Dataset	Methodology	Feature selection	XAI	Performance results (%)
[35]	X Dataset (by X API)	(IT2-M) (FIS) (IT2-S) (FIS) (IT1-M) (FIS) (IT1-S) (FIS)	○	○	Acc = 95.5 P = 95.7 F = 96.2 R = 96.7 AUC = 97.1
[36]	SemCat-(2018)	TOBEAT leveraging BERT and CNN	●	○	Acc = 94.97 P = 94.05 R = 95.88 F = 94.95
[37]	X Dataset (by hatebase.org)	A clustering framework using probabilistic rules and fuzzy sentiment classification	○	○	Acc = 94.53 P = 92.54 R = 91.74 F = 92.56 AUC = 96.45
[47]	X Dataset (Sem-Eval-2014 and Bamman)	Classification methodology based on (FL)	●	○	Acc = 90.9 P = 95.7 R = 82.4 F = 87.4
[48]	X Dataset (by [48])	Ensemble Learning technique with Random oversampling (ROS) plus random undersampling (RUS) plus fuzzy-based oversampling (FOS)	○	○	Mean P= 0.76-0.78 Mean F= 0.76-0.55 Mean FP=0.11 TP= 0.74-0.43
[39]	X Dataset (by X API)	(HMPS) Hierarchical Meta-Path Based Approach with Feedback and default one-class classifier	○	○	P = 95.0 R = 90.0 F = 93.0 AUC = 92.0
[49]	X Dataset (by X API)	Deep Learning(DL) Methodology Utilizing a Multilayer Perceptron (MLP) algorithm	○	○	P = 92.0 R = 88.0 F = 89.0

 doi:10.20944/preprints202509.0410.v1

[34]	X Dataset (by www.unipi.it)	Ensemble based XGBoost with Random Forest	●	●	Acc = 90 P = 91.0 R = 86.0 F = 89.0
[50]	X Dataset (HSpam14 and 1KS10KN)	Ensemble Method Utilizing Convolutional Neural Network Models and a Feature-Based Model	○	○	Acc = 95.7 P = 92.2 R = 86.7 F = 89.3
[51]	X Dataset (by [49])	LR,SVM and RF utilizing various character N-gram features.	○	○	P = 79.5 R = 79.4 F = 79.4
Proposed RFO-AdaBoost	X Dataset (by [35])	nature-inspired with ensemble Learning approach	●	●	Acc = 99.35 P = 99.03 R = 99.86 F = 99.44

This study attains an accuracy of 99.35%, exceeding the results documented in comparable works. Significantly, this boost in efficiency can be ascribed to our holistic methodological framework. This encompasses the execution of a preprocessing method, Adapt RFO nature-inspired algorithm for feature selection which for the first time applied to cybersecurity.

Additionally, implementation of XAI strategy to enhance interpretability of the model. Compared to the related studies which neither utilized advantage of optimization features selection algorithms or harnessed the capabilities of XAI or both. Our methodology illustrates the substantial influence for such techniques on the effectiveness of spam identification. Employment of the meta-learning along with nature-inspired optimizer feature selection method provides a strong framework adept at managing the intricacies involved for spam classification problem. Furthermore, applying XAI approaches facilitates a more profound comprehension for how the model makes decisions, which increases model predictions understand and validation. This represents an important difference with previous methodologies mentioned, especially those which depend on conventional "black box" ML methods that lack providing clarity about their working mechanisms. Moreover, integration of XAI enhances the performance of our detection method and establishes a foundation to develop interpreted cybersecurity-based AI solutions. Consequently, this research not only enhances performance score within the discipline yet propels the discussion around the impact of nature-inspired meta-heuristic method with interpreted artificial intelligence models, establishing an additional benchmark for subsequent investigations in spam identification and related fields.

6. Conclusions

This research introduced a lightweight robust XAI-powered AdaBoost ensemble learning for X.com spam account detection. The proposed architecture consists of four components: preprocessing the data, feature selection, spam classification, and explanation of model prediction. Firstly, the preprocessing data. Following that, the dataset is handled by RFO which selects a subset

 doi:10.20944/preprints202509.0410.v1

of nine features of spam accounts dataset. Machine learning algorithms are trained and evaluated using the subset feature. The model's prediction is ultimately interpreted by the application of Shapley values. The abilities and shortcomings of the proposed model are put into perspective by comparing the AdaBoost ensemble learning , which was selected for its adaptive learning capability and explainability, to well-known machine learning algorithms, such as DT, KNN and LR. Importantly, the proposed system outstanding results with an accuracy of roughly 99.35 %, and leverages Swarm and summary Shapley to clarify its process for making decisions, hence improving model interpretability. The research results providing a clear technique that improves interpret among complex environments such as cybersecurity.

These results stress the pivotal importance of optimal feature subset selection and explainability AI, necessitating additional investigation into nature-inspired optimization algorithms and XAI methodologies.

Moreover, the proposed approach has limitations. The efficacy of the proposed system is intricately linked to the spam accounts dataset characteristics, that might not accurately represent the discrepancies found in alternative datasets. in conjunction with the intrinsic constraints of the AdaBoost and optimization algorithm. The estimations employed by SHAP, underscores the intricate equilibrium among performance, explainability and generalizability. The forthcoming work motivated by our results seems motivated and essential. We intend to evaluate the versatility of the proposed approach across other OSN platforms. Furthermore, rectifying the approach constraints found in this study, future study will explore different feature selection methodologies and the efficacy of ensemble learning approaches to improve efficiency and explainability.

References

1. G. Jethava and U. P. Rao, "Exploring security and trust mechanisms in online social networks: An extensive review," *Computers & Security,* p. 103790, 2024.
2. D. Nevado-Catalán, S. Pastrana, N. Vallina-Rodriguez, and J. Tapiador, "An analysis of fake social media engagement services," *Computers & Security,* vol. 124, p. 103013, 2023.
3. E. de Keulenaar, J. C. Magalhães, and B. Ganesh, "Modulating Moderation: A Genealogy of Objectionable Content on Twitter," *MediArXiv. September,* vol. 1, 2022.
4. R. Murtfeldt, N. Alterman, I. Kahveci, and J. D. West, "RIP Twitter API: A eulogy to its vast research contributions," *arXiv preprint arXiv:2404.07340,* 2024.
5. J. Song, S. Lee, and J. Kim, "Spam filtering in twitter using sender-receiver relationship," in *Recent Advances in Intrusion Detection: 14th International Symposium, RAID 2011, Menlo Park, CA, USA, September 20-21, 2011. Proceedings 14*, 2011: Springer, pp. 301-317.
6. S. J. Delany, M. Buckley, and D. Greene, "SMS spam filtering: Methods and data," *Expert Systems with Applications,* vol. 39, no. 10, pp. 9899-9908, 2012.
7. Z. Iman, S. Sanner, M. R. Bouadjenek, and L. Xie, "A longitudinal study of topic classification on twitter," in *Proceedings of the International AAAI Conference on Web and Social Media*, 2017, vol. 11, no. 1, pp. 552-555.
8. H. Rashid, H. B. Liaqat, M. U. Sana, T. Kiren, H. Karamti, and I. Ashraf, "Framework for detecting phishing crimes on Twitter using selective features and machine learning," *Computers and Electrical Engineering,* vol. 124, p. 110363, 2025.
9. S. B. S. Ahmad, M. Rafie, and S. M. Ghorabie, "Spam detection on Twitter using a support vector machine and users' features by identifying their interactions," *Multimedia Tools and Applications,* vol. 80, no. 8, pp. 11583-11605, 2021.
10. S. Bazzaz Abkenar, E. Mahdipour, S. M. Jameii, and M. Haghi Kashani, "A hybrid classification method for Twitter spam detection based on differential evolution and random forest," *Concurrency and Computation: Practice and Experience,* vol. 33, no. 21, p. e6381, 2021.
11. A. Galli, V. La Gatta, V. Moscato, M. Postiglione, and G. Sperlì, "Explainability in AI-based behavioral malware detection systems," *Computers & Security,* vol. 141, p. 103842, 2024.
12. T.-T.-H. Le, H. Kim, H. Kang, and H. Kim, "Classification and explanation for intrusion detection system based on ensemble trees and SHAP method," *Sensors,* vol. 22, no. 3, p. 1154, 2022.

 doi:10.20944/preprints202509.0410.v1

13. M. Braik and H. Al-Hiary, "Rüppell's fox optimizer: A novel meta-heuristic approach for solving global optimization problems," *Cluster Computing,* vol. 28, no. 5, pp. 1-77, 2025.
14. K. Kaczmarek-Majer *et al.*, "PLENARY: Explaining black-box models in natural language through fuzzy linguistic summaries," *Information Sciences,* vol. 614, pp. 374-399, 2022.
15. A. A. Khan, O. Chaudhari, and R. Chandra, "A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation," *Expert Systems with Applications,* vol. 244, p. 122778, 2024.
16. A. Mohammed and R. Kora, "A comprehensive review on ensemble deep learning: Opportunities and challenges," *Journal of King Saud University-Computer and Information Sciences,* vol. 35, no. 2, pp. 757-774, 2023.
17. H.-J. Xing, W.-T. Liu, and X.-Z. Wang, "Bounded exponential loss function based AdaBoost ensemble of OCSVMs," *Pattern Recognition,* vol. 148, p. 110191, 2024.
18. E. M. Ferrouhi and I. Bouabdallaoui, "A comparative study of ensemble learning algorithms for high-frequency trading," *Scientific African,* vol. 24, p. e02161, 2024.
19. G. Paic and L. Serkin, "The impact of artificial intelligence: from cognitive costs to global inequality," *EUROPEAN PHYSICAL JOURNAL-SPECIAL TOPICS,* 2025.
20. A. Abusitta, M. Q. Li, and B. C. Fung, "Survey on Explainable AI: Techniques, challenges and open issues," *Expert Systems with Applications,* vol. 255, p. 124710, 2024.
21. E. Borgonovo, E. Plischke, and G. Rabitti, "The many Shapley values for explainable artificial intelligence: A sensitivity analysis perspective," *European Journal of Operational Research,* vol. 318, no. 3, pp. 911-926, 2024.
22. V. Selvakumar, N. K. Reddy, R. S. V. Tulasi, and K. R. Kumar, "Data-Driven Insights into Social Media Behavior Using Predictive Modeling," *Procedia Computer Science,* vol. 252, pp. 480-489, 2025.
23. I. D. Mienye and N. Jere, "A survey of decision trees: Concepts, algorithms, and applications," *IEEE access,* 2024.
24. S. Mohammed, N. Al-Aaraji, and A. Al-Saleh, "Knowledge Rules-Based Decision Tree Classifier Model for Effective Fake Accounts Detection in Social Networks," *International Journal of Safety & Security Engineering,* vol. 14, no. 4, 2024.
25. R. K. Halder, M. N. Uddin, M. A. Uddin, S. Aryal, and A. Khraisat, "Enhancing K-nearest neighbor algorithm: a comprehensive review and performance analysis of modifications," *Journal of Big Data,* vol. 11, no. 1, p. 113, 2024.
26. M. Teke and T. Etem, "Cascading GLCM and T-SNE for detecting tumor on kidney CT images with lightweight machine learning design," *The European Physical Journal Special Topics,* pp. 1-16, 2025.
27. Q. Ouyang, J. Tian, and J. Wei, "E-mail Spam Classification using KNN and Naive Bayes," *Highlights in Science, Engineering and Technology,* vol. 38, pp. 57-63, 2023.
28. E. Bisong and E. Bisong, "Logistic regression," *Building machine learning and deep learning models on google cloud platform: A comprehensive guide for beginners,* pp. 243-250, 2019.
29. S. K. Sarker *et al.*, "Email Spam Detection Using Logistic Regression and Explainable AI," in *2025 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, 2025: IEEE, pp. 1-6.
30. K. K. Bharti and S. Pandey, "Fake account detection in twitter using logistic regression with particle swarm optimization," *Soft Computing,* vol. 25, no. 16, pp. 11333-11345, 2021.
31. S. B. Abkenar, M. H. Kashani, M. Akbari, and E. Mahdipour, "Learning textual features for Twitter spam detection: A systematic literature review," *Expert Systems with Applications,* vol. 228, p. 120366, 2023.
32. A. Qazi, N. Hasan, R. Mao, M. E. M. Abo, S. K. Dey, and G. Hardaker, "Machine Learning-Based Opinion Spam Detection: A Systematic Literature Review," *IEEE Access,* 2024.
33. N. H. Imam and V. G. Vassilakis, "A survey of attacks against twitter spam detectors in an adversarial environment," *Robotics,* vol. 8, no. 3, p. 50, 2019.
34. E. Alnagi, A. Ahmad, Q. A. Al-Haija, and A. Aref, "Unmasking Fake Social Network Accounts with Explainable Intelligence," *International Journal of Advanced Computer Science & Applications,* vol. 15, no. 3, 2024.

 doi:10.20944/preprints202509.0410.v1

35. İ. Atacak, O. Çıtlak, and İ. A. Doğru, "Application of interval type-2 fuzzy logic and type-1 fuzzy logic-based approaches to social networks for spam detection with combined feature capabilities," *PeerJ Computer Science,* vol. 9, p. e1316, 2023.
36. S. Ouni, F. Fkih, and M. N. Omri, "BERT-and CNN-based TOBEAT approach for unwelcome tweets detection," *Social Network Analysis and Mining,* vol. 12, no. 1, p. 144, 2022.
37. F. E. Ayo, O. Folorunso, F. T. Ibharalu, I. A. Osinuga, and A. Abayomi-Alli, "A probabilistic clustering model for hate speech classification in twitter," *Expert Systems with Applications,* vol. 173, p. 114762, 2021.
38. S. Liu, Y. Wang, C. Chen, and Y. Xiang, "An ensemble learning approach for addressing the class imbalance problem in Twitter spam detection," in *Information Security and Privacy: 21st Australasian Conference, ACISP 2016, Melbourne, VIC, Australia, July 4-6, 2016, Proceedings, Part I 21*, 2016: Springer, pp. 215-228.
39. S. Gupta, A. Khattar, A. Gogia, P. Kumaraguru, and T. Chakraborty, "Collective classification of spam campaigners on Twitter: A hierarchical meta-path based approach," in *Proceedings of the 2018 world wide web conference*, 2018, pp. 529-538.
40. P. Manasa *et al.*, "Tweet spam detection using machine learning and swarm optimization techniques," *IEEE Transactions on Computational Social Systems,* 2022.
41. R. Krithiga and E. Ilavarasan, "Hyperparameter tuning of AdaBoost algorithm for social spammer identification," *International Journal of Pervasive Computing and Communications,* vol. 17, no. 5, pp. 462-482, 2021.
42. A. Ghourabi and M. Alohaly, "Enhancing spam message classification and detection using transformer-based embedding and ensemble learning," *Sensors,* vol. 23, no. 8, p. 3861, 2023.
43. R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*, 1995, vol. 14, no. 2: Montreal, Canada, pp. 1137-1145.
44. R. E. Schapire, "Explaining adaboost," in *Empirical inference: festschrift in honor of vladimir N. Vapnik*: Springer, 2013, pp. 37-52.
45. M. Djuric, L. Jovanovic, M. Zivkovic, N. Bacanin, M. Antonijevic, and M. Sarac, "The adaboost approach tuned by sns metaheuristics for fraud detection," in *Proceedings of the International Conference on Paradigms of Computing, Communication and Data Sciences: PCCDS 2022*, 2023: Springer, pp. 115-128.
46. F. Jáñez-Martino, R. Alaiz-Rodríguez, V. González-Castro, E. Fidalgo, and E. Alegre, "A review of spam email detection: analysis of spammer strategies and the dataset shift problem," *Artificial Intelligence Review,* vol. 56, no. 2, pp. 1145-1173, 2023.
47. A. B. Meriem, L. Hlaoua, and L. B. Romdhane, "A fuzzy approach for sarcasm detection in social networks," *Procedia Computer Science,* vol. 192, pp. 602-611, 2021.
48. S. Liu, Y. Wang, J. Zhang, C. Chen, and Y. Xiang, "Addressing the class imbalance problem in twitter spam detection using ensemble learning," *Computers & Security,* vol. 69, pp. 35-49, 2017.
49. A. K. Ameen and B. Kaya, "Spam detection in online social networks by deep learning," in *2018 international conference on artificial intelligence and data processing (IDAP)*, 2018: IEEE, pp. 1-4.
50. S. Madisetty and M. S. Desarkar, "A neural network-based ensemble approach for spam detection in Twitter," *IEEE Transactions on Computational Social Systems,* vol. 5, no. 4, pp. 973-984, 2018.
51. M. Ashour, C. Salama, and M. W. El-Kharashi, "Detecting spam tweets using character N-gram features," in *2018 13th International conference on computer engineering and systems (ICCES)*, 2018: IEEE, pp. 190-195.