Baghdad Science Journal

Volume 23 | Issue 3 Article 27

3-25-2026

Enhanced Botnet Detection Using a Modified Naïve Bayes Algorithm with Laplace Smoothing and FAMD-Based Feature Agglomeration

Ahmed L. Alshami
Department of Computer Science, College of Science, Mustansiriyah University, Baghdad, Iraq, ahmed.alshami@uomustansiriyah.edu.iq

Bashar M. Nema
Department of Computer Science, College of Science, Mustansiriyah University, Baghdad, Iraq, bmn774@uomustansiriyah.edu.iq

Alaa H. Al-Hamami
Department of Cybersecurity, Dijla University College, Baghdad, Iraq, alaa.hussein@alshaab.edu.iq

How to Cite this Article

Alshami, Ahmed L.; Nema, Bashar M.; and Al-Hamami, Alaa H. (2026) "Enhanced Botnet Detection Using a Modified Naïve Bayes Algorithm with Laplace Smoothing and FAMD-Based Feature Agglomeration," *Baghdad Science Journal*: Vol. 23: Iss. 3, Article 27.
DOI: https://doi.org/10.21123/2411-7986.5252

BAGHDAD SCIENCE JOURNAL 2026;23(3):1057–1067

Scan the QR to view the full-text article on the journal website

RESEARCH ARTICLE

Enhanced Botnet Detection Using a Modified Naïve Bayes Algorithm with Laplace Smoothing and FAMD-Based Feature Agglomeration

Ahmed L. Alshami [1], **Bashar M. Nema** [1,*], **Alaa H. Al-Hamami** [2]

[1] Department of Computer Science, College of Science, Mustansiriyah University, Baghdad, Iraq
[2] Department of Cybersecurity, Dijla University College, Baghdad, Iraq

ABSTRACT

Botnets turned into a security problem that would put user privacy and security at risk. Progressive and flexible Machine Learning (ML) techniques are necessary for robust botnet revealing. In this study researchers presented an integration of the modified Naïve Bayes and M3L algorithms that uses Factor Analysis of Mixed Data (FAMD) for feature aggregation, and Laplace smoothing for adjustment to address the limitation of conventional Naïve Bayes classifiers for detecting botnet. The modified algorithm uses Laplace smoothing to address problems with zero-frequency data and improve the classifier's adaptability. FAMD was used to merge continuous and categorical features to improve the algorithm's capacity and handle mixed data types and lowering the feature dimensionality. Better classification performance and less computing complexity follow from this. Through using a dataset of network traffic that includes both benign and botnet, the proposed approach was evaluated. The suggested algorithm achieves a 97.45% accuracy and a Mean Squared Error (MSE) of 0.039. These results show the ability of the combined method in accurately detect botnet activity and reduce related security threats.

Keywords: Botnet, Factor Analysis of Mixed Data (FAMD), Naïve Bayes (NB), M3L, Internet of Things (IoT), Machine Learning (ML)

Introduction

Botnets which are networks that have compromised computers controlled by an attacker and became a major security threat nowadays, these networks are often used to initiate Distributed Denial of Service (DDoS) attacks, spread malware, or steal sensitive data, which can cause serious financial and reputational damages to individuals, organizations, and governments. Due to the complexity growth of botnet activities and the developing their hiding techniques, classical signature-based as well as anomaly-based methods of detection fails to keep up, rising the need for developing more advanced and adaptive machine learning techniques for better detection and reduction of botnets. Naïve Bayes classifiers became widely used in multiple cybersecurity solutions due to simplicity,[1–3] computational efficiency, and capability to handle large datasets. Yet, these classifiers are based on the assumption of independence among features, which is often not applied in real-world datasets, leading to limited performance. Moreover, Naïve Bayes classifiers are sensitive to noisy or irrelevant features and can suffer from issues related to zero-frequency data, impacting their generalization capabilities. Botnet forensics is a sub- discipline of digital forensics focused on the investigation and analysis of botnet activities and infrastructure.[4] It involves the identification of bot-infected hosts, Command and Control (C&C) servers Fig. 1, and the

Received 27 October 2023; revised 1 July 2024; accepted 3 July 2024.
Available online 25 March 2026

* Corresponding author.
E-mail addresses: ahmed.alshami@uomustansiriyah.edu.iq (A. L. Alshami), bmn774@uomustansiriyah.edu.iq (B. M. Nema), alaa.hussein@alshaab.edu.iq (A. H. Al-Hamami).

https://doi.org/10.21123/2411-7986.5252

analysis of botnet communication patterns to understand their structure and behavior.[5] Various machine learning algorithms have been applied to botnet forensics, including Support Vector Machines (SVM), Decision Trees, and Naïve Bayes.[6] The Naïve Bayes classifier is a probabilistic machine learning algorithm based on Bayes' theorem, which assumes that features are conditionally independent given the class label.[7] Even with its uncomplicated structure, Naïve Bayes classifier showed notable performance in multiple applications, which can be text classification, spam filtering, and network intrusion detection.[8] Laplace smoothing is a technique that is used to treat the zero-frequency problem in the Naïve Bayes algorithm, since a probability of zero is given for feature that does not occur in the dataset used for training, as a result the entire conditional probability to be zero.[9] Therefore, adding a constant value to the count of each feature, allows Laplace smoothing to prevent zero probabilities and offers better estimations, especially for limited training data.[10] Feature agglomeration is a dimensionality reduction technique involves clustering for alike features to groups then replace them with a one representative feature.[11] This way helps reduce noise, improve computational efficiency, and address the course of dimensionality in machine learning algorithms.[12]

Botnets can compromise an IoT-based system through a different kind of techniques, along with exploiting weaknesses in the devices, using brute force attacks to find weak passwords, or mislead users into installing malware. After botnet gains access to any IoT device, it may control the device and apply a variety of malicious activities. One usual technique used by botnets is to launch Distributed Denial of Service (DDoS) attacks, in which many devices are used to flood any target server or network with traffic, causing it to became overwhelmed and not accessible to legitimate users. Botnets could also spread malware, access and steal sensitive information, or use the infected devices to mine cryptocurrency. The issue with IoT devices is that they are typically designed to be low-cost and low-power,[14] with limited processing and memory capabilities. Therefore, they could lack basic security features, like secure boot and firmware updates, or encryption of sensitive data. This makes them an easy target for botnets and other attackers. Also, because IoT devices are usually connected to the internet through a router or gateway, which likely have vulnerabilities that can be in benefit of botnets to gain access to the devices. After a botnet breaches a device, it can use it as a base to spread towards other devices in the same network, expanding and leading to more difficulty to detect and remove. Protecting IoT-based systems from botnet incursion, it is important to take a vast tactic for security, including regular software updates, strong password policies, network segmentation, and the use of intrusion detection and prevention systems. Additionally, IoT devices developers better concentrate on security in their designs to ensure upgradable devices which can receive security updates.

This paper intends to overcome previously mentioned limitations by proposing a modified Naïve Bayes algorithm to detect botnet. This algorithm includes Laplace Smoothing and Factor Analysis of Mixed Data (FAMD) for feature clustering. The contributions can be briefed by: A modified Naïve Bayes

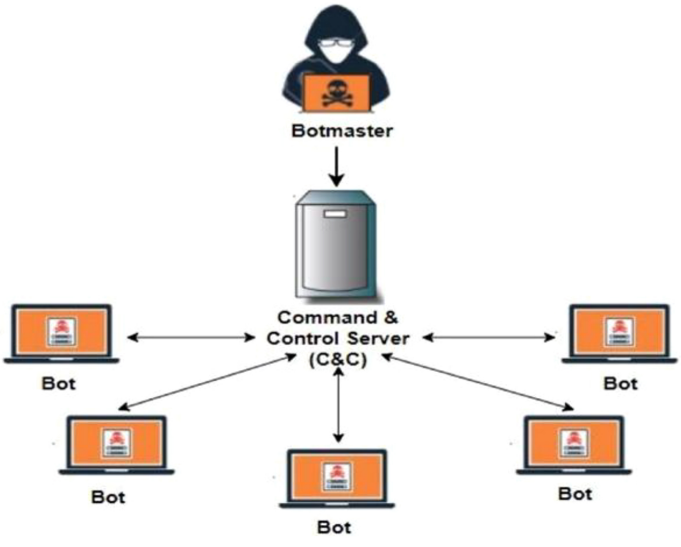

Fig. 1. Botnet architecture through C&C.[13]

classifier that leverages Laplace Smoothing solving issues associated with zero-repeat data and improve the classifier's ability. This improvement ensures the algorithm remains effective even in rare or unfamiliar events. Dealing with the challenges arising from data type diversity and high dimensionality, FAMD is used for feature clustering, enabling the algorithm to successfully handle continuous and categorical features. This approach reduces the dimensionality of the feature space and elevates the classifier's ability to handle noisy or unrelated features, resulting in better classification performance and reduced computational complexity. The proposed technique was well tested on a network traffic dataset containing both normal and botnet records. The results show that the modified Naïve Bayes algorithm exceeds traditional classifiers of the same type and other recent ML techniques, achieving a notable accuracy of 98% and a Mean Squared Error (MSE) of 0.05. As a result of these contributions, this paper demonstrates the potential of the modified Naïve Bayes algorithm, using Laplace Smoothing and FAMD based feature clustering, as an effective tool for botnet detection. By addressing the limitations of traditional Naïve Bayes classifiers, the proposed approach present notable improvements in accuracy and generalization capabilities, contributing to the development of more robust and scalable cybersecurity solutions.

Materials and methods

Naïve bayes algorithm

The Naïve Bayes algorithm is a basic and commonly used probabilistic classification technique based on Bayes' theorem, which is used for various machine learning tasks.[15] The fundamental understanding of the Naïve Bayes algorithm is that the features are conditionally independent given the class label. In other words, the occurrence of each feature is assumed to be independent of the others when the class label is known. Given a dataset with n features (X1, X2, …, Xn) and a class label C, the Naïve Bayes classifier aims to predict the probability of a particular class label given a set of features. According to Bayes' theorem, this probability can be expressed as in Eq. (1):

$$(C|X1, X2, \ldots, Xn) = (C) * (X1, X2, \ldots, Xn|C) / P(X1, X2, \ldots, Xn) \quad (1)$$

Due to the conditional independence assumption, the joint probability P(X1, X2, …, Xn|C) can be simplified as in Eq. (2), and the classifier assigns the class label with the highest probability to a given instance:

$$(X1, X2, \ldots, Xn|C) = (X1|C) * (X2|C) * \ldots * P(Xn|C) \quad (2)$$

Laplace Smoothing: Laplace smoothing, also known as additive smoothing, is a technique employed to address the issue of zero- frequency data in Naïve Bayes classifiers. When a feature-class combination has not been observed in the training data, the probability estimate for that combination becomes zero, which can significantly affect the classifier's performance. Laplace smoothing addresses this issue by adding a constant α (typically 1) to the count of each feature-class combination, ensuring that no probability estimate is zero. The first proposed step of modified smoothed probability estimate can be calculated as Eq. (3):

$$(Xi|C) = ((Xi, C) + \alpha) / ((C) + \alpha * |V|) \quad (3)$$

where N(Xi, C) is the count of instances with feature Xi and class label C, N(C) is the total count of instances with class label C, |V| is the number of distinct values of feature Xi, and α is the smoothing constant.

Factor analysis of mixed data (FAMD)

Factor Analysis of Mixed Data (FAMD) is an extension of principal component analysis (PCA) that allows the simultaneous analysis of continuous and categorical variables.[14] FAMD aims to find hidden factors or dimensions that represent the underlying structure and relationships in the data, so that reducing the dimensionality of the dataset while maintaining as much information as possible, see Fig. 2. FAMD works by optimizing a weighted combination of continuous and categorical variables to maximize the explained variance.

Factor Analysis of Mixed Data (FAMD) is a dimensionality reduction method that allows for the simultaneous analysis of continuous and categorical variables in a dataset. FAMD is an extension of Factor Analysis (FA), which is a statistical method used to identify hidden variables, or factors, that explain the variation in a set of detected variables. FAMD can deal with mixed data types, including nominal, ordinal, and continuous variables, and can reveal the underlying structure in the data while reducing the dimensionality. This makes it a useful tool for data exploration and visualization, as well as for building predictive models. The FAMD algorithm begins by

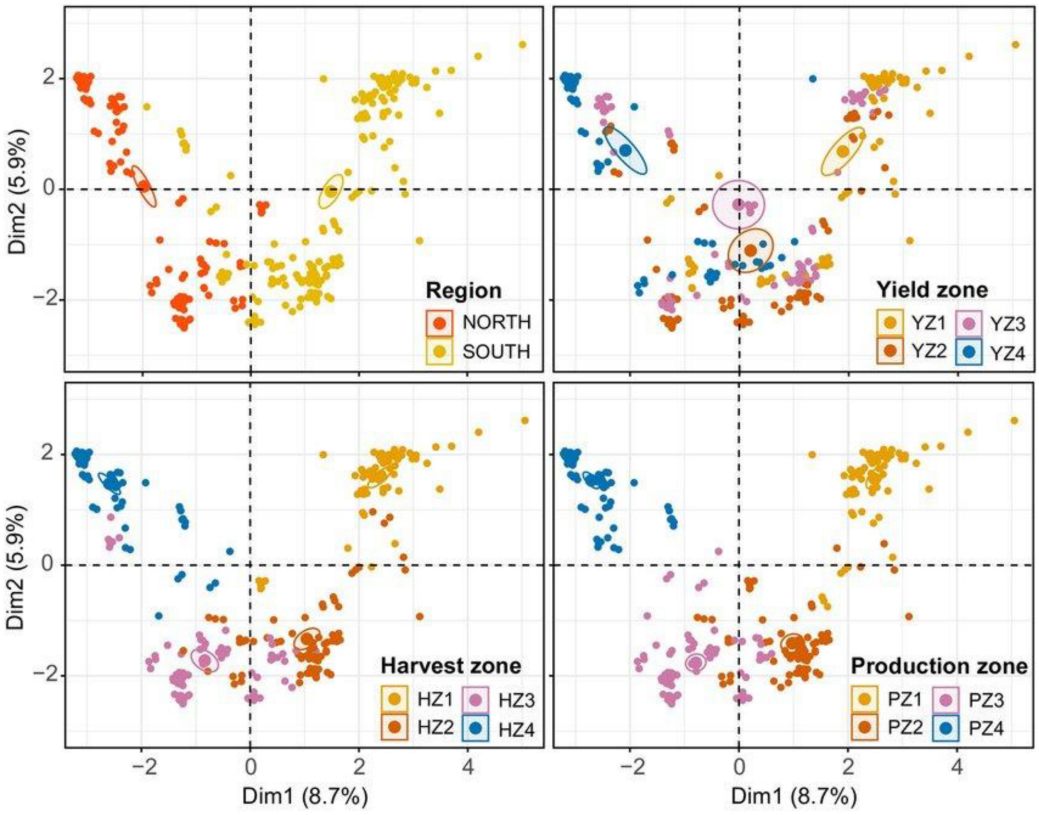

Fig. 2. Dimensionality reduction in FAMD.[15]

computing a matrix of dissimilarities between the variables in the dataset, based on their data types. For categorical variables, the dissimilarity measure is based on the chi-square distance, while for continuous variables, it is based on the Euclidean distance. following, (FAMD) applies Singular Value Decomposition (SVD) to the contrast matrix, breaking down the matrix into a set of independent factors that carry the most differences in the data. Later these factors are ordered depending on their importance, with the first factor explaining the largest portion of the variance, while following factors capture smaller portions. The factor loadings are then used to compute new scores for each observation in the dataset, representing their position in the factor space. These scores can be used for data visualization and exploration, as well as for building predictive models. One of the advantages of FAMD is that it can handle missing data and can impute missing values based on the observed data. This could enhance the accuracy of the factor analysis and minimize the risk of bias.[16]

The max-margin multi-label classification primal formulation (M3L)

The purpose of multi-label classification is to learn a function f that may be utilized to attach multiple labels to a point x. Assumption that N training data points had been provided in the form (xi, yi) ∈ RD × {±1}L, where yil is +1 if label l has been assigned to point i, and –1 otherwise. Note that such an encoding enables us to get information from both the present and missing labels, since both can provide useful information while making predictions for the test categories. A principled approach to stating the issue would involve minimizing the loss function, which is of utmost importance, over the training set while considering regularization or past information. Indeed, due to the inherent difficulty in directly minimizing most discrete loss functions, instead it is possible to minimize an upper bound on the loss, such as the hinge loss. The learning problem can be formulated as the primal problem.[17,18], Eq. (4):

$$P_i = \min_f \frac{1}{2}\|f\|^2 + C\sum_{i=1}^{N}\xi_i \tag{4}$$

$$s.t. f(x_i, y_i) \geq f(x_i, y) + \Delta(y_i, y) - \xi_i \quad \forall i, y \in \{\pm 1\}^L \setminus \{y_i\} \quad \forall i, \xi_i \geq 0$$

The labels for the new point x are assigned based on the maximum value of the function f(x, y). The disadvantage of this formulation is that it involves $N2^L$ restrictions, which significantly impede direct optimization, resulting in poor performance. In

addition, the process of categorizing new data points may need 2^L function evaluations, with each evaluation corresponding to a potential value of y. This can be impractical during runtime. In this section, establishing that, assuming linearity, (P1) may be restated as the minimization of L densely coupled sub-problems, each with N constraints. Simultaneously, the cost of prediction is reduced to a single function evaluation, with a complexity that is directly proportional to the number of labels. The concepts that formed the basis of this deconstruction were employed in a multitask learning scenario. Nonetheless, their aim is to amalgamate several activities into a solitary learning issue, whereas our focus lies in breaking down (3) into multiple subproblems.

The M3L dual has similarities to the conventional SVM dual. Consequently, existing optimization approaches may be applied. However, the deep structure of R couples all NL dual variables and just transferring current methods leads to incredibly inefficient code. The effectively transition from an algorithm having a time complexity of $O(L^2)$ to an algorithm with a time complexity of O(L). In addition, our techniques may achieve high efficiency for non-linear problems by reusing the kernel cache. Researchers handling the kernelized and linear M3L situations as distinct entities. [19,20]

Botnet

A botnet is a network of compromised computers or devices, also known as bots, which are remotely controlled by an attacker, typically without the knowledge of their owners. These botnets can be used to carry out a variety of malicious activities, such as launching Distributed Denial of Service (DDoS) attacks, spreading malware, stealing sensitive information, or conducting other cybercrimes. The primary challenge in combating botnets is the detection and identification of botnet activities within network traffic, as the attackers often employ sophisticated techniques to obfuscate their activities and blend in with legitimate traffic. Botnets are networks contains infected devices or computers that are directed by an attacker, referred as a botmaster. Botnets are a useful tool to launch different kinds of malicious actions, that includes Distributed Denial of Service (DDoS), spamming, possessing unauthorized confidential data, and cryptocurrency mining. [16] When any device is infected by a botnet, it became one within a larger network of devices that are managed by the botmaster. The botmaster could start commands to the devices, which will then carry out the instructions beyond user's awareness or consent. Detecting botnets might be difficult since they are designed to be invisible to stay undetected. Even though, attackers may create detectable signs that can be used for tracing, such as: [21,22]

1. Network traffic: Botnets usually produce great amount of network traffic, which could be tracked and studied for any signs of botnet presence. For example, DDoS attacks usually include large number of devices sending heavy data flow to a single target, causing it to overload and offline.
2. Communication protocols: Botnets regularly use particular communication protocols or ports to contact with the botmaster or another infected devices. Analyzing network traffic and its pattern which can assist spotting botnet behavior.
3. Command and control servers: Botnets mostly depend on command and control servers to send instructions to the compromised devices. By recognizing these servers and tracking their activity, it may be possible to detect botnet activity.
4. Behavioral patterns: Botnets might show specific behavioral patterns that are used for detection, such as logging to specific websites or resources, or using precise system resources.

To detect botnets accurately, it is preferable to combine several techniques, including network monitoring, anomaly detection, and machine learning algorithms. By examining network traffic, communication protocols, C&C, and behavioral patterns, it is possible to notify botnets and reduce their effect on computer networks. Fig. 3 shows the Botnet detection process.

Proposed method

In this section, the proposed method is introduced for detecting the botnet using a modified Naïve Bayes algorithm that uses Laplace smoothing and Factor Analysis of Mixed Data (FAMD) for feature agglomeration. The process follows four main steps: data preprocessing, feature agglomeration using FAMD, modified Naïve Bayes algorithm with Laplace smoothing classifier, M3L with Laplace smoothing classifier, and performance evaluation.

Data preprocessing

The first step in proposed method is data preprocessing, which involves cleaning and formatting the raw network data into an appropriate format for the modified Naïve Bayes and M3L algorithms. This step includes removing any unnecessary or noisy features,

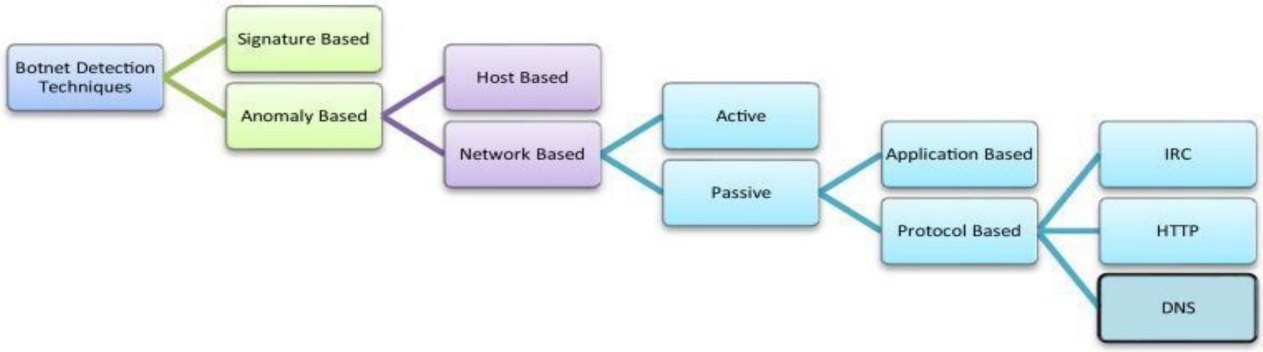

Fig. 3. Process of botnet detection.[21]

Table 1. The N-BaloT dataset for IOT botnet detection.[18]

Aggregated by	Value	Statistics	Total No. of features
Source IP	Packet size (only outbound)	Mean, variance	3
	Packet count	Integer	
Source MAC-IP	Packet size (only outbound)	Mean, variance	3
	Packet count	Integer	
Channel	Packet size (only outbound)	Mean, variance	10
	Packet count	Integer	
	Amount of time between packet arrivals	Mean, variance, integer	
	Packet size (both inbound and outbound)	Magnitude, radius, covariance, correlation coefficient	
Socket	Packet size (only outbound)	Mean, variance	7
	Packet size (both inbound and outbound)	Magnitude, radius, covariance, correlation coefficient	

handling missing values, and normalizing continuous features to secure that they are on the same scale.

The N-BaIoT dataset is an available public dataset that was collected specifically for botnet detection in IoT- based systems. The dataset includes network traffic records generated by 12 different types of IoT devices, including IP cameras, smart TVs, smart home appliances, beside other devices. The devices were intentionally compromised with multiple botnets to simulate real-world botnet behavior. The dataset contains a total of 50,000 network traffic records, with 25,000 records for benign activity and 25,000 records for botnet activity. The records were collected over a period of several days and capture various features relevant to botnet detection, such as packet size, packet length, frequency, source and destination IP addresses, protocol, and time stamps. The N-BIoT dataset for IoT botnet detection shown in Table 1.

The dataset includes labeled instances, with each instance being labeled as either benign or botnet based on the type of botnet that infected the device. The dataset includes five different types of botnets, including Mirai, Bashlite, Gafgyt, BASHLITE, and Tsunami. The N-BaIoT dataset also contains 17 different features which were selected upon their importance for botnet detection in IoT-based systems. These features include packet length mean, packet length variance, flow duration, protocol, source port, destination port, and others. The dataset was designed to assist the development and testing of machine learning algorithms for botnet detection in IoT-based systems. The labeled data make it possible to train and evaluate of supervised learning algorithms, while the feature set allows for the exploration and visualization of the dataset.

The dataset contains a mix of normal and botnet examples to enable the training and evaluation of machine learning algorithms used for botnet detection. The features in the dataset include:

- Device type: This feature recognizes the type of IoT device, such as a smart home thermostat, camera, or refrigerator.
- Network traffic: This feature records the network data flow produced by the device, including packet size, packet length, packet frequency, and the types of packets (e.g., HTTP, TCP, UDP). Source and destination.
- IP addresses: This feature identifies the source and destination IP addresses of the network traffic.
- Protocol: This feature identifies the protocol used by the network traffic (e.g., HTTP, HTTPS, SMTP).
- Time stamps: This feature record the time stamps for the data flow, which can be applied in time-series analysis and irregular behavior detection. Geo-location: This feature identifies the

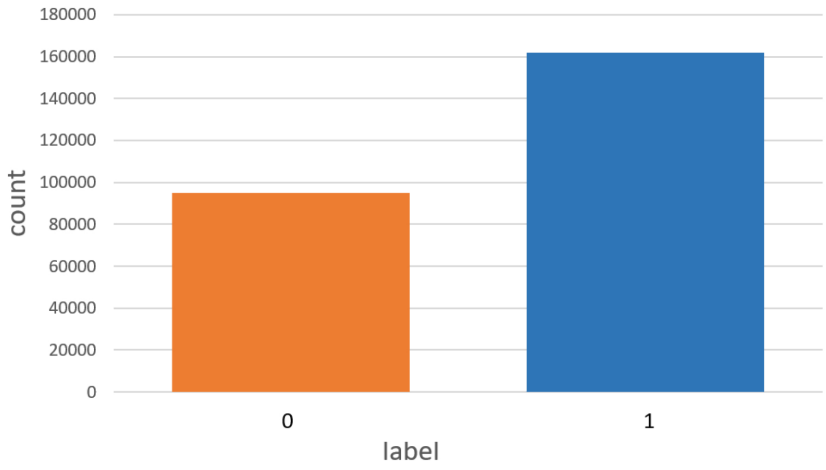

Fig. 4. Attribute label plot.

geographic location of the device based on its IP address.
- DNS queries: This feature records the DNS quests sent by the device, which can be used to identify harmful domains.
- Firmware version: This feature determines the firmware version of the device, which can be used to reveal weaknesses that botnet may take advantage of.
- Authentication: This feature remarks the authentication method used by the device, like a password or certificate, and can be used to find out weak authentication methods.

The dataset also provides labels that identify each record as normal or botnet, based on manual review or known botnet patterns. This labeling can help to train and test machine learning algorithms for botnet detection. Overall, a dataset for botnet detection in IoT devices need to cover a range of features that are important to identifying botnet activity in network traffic and include both benign and botnet instances to make it possible for the development and evaluation of powerful machine learning algorithms. The attribute label plot shown in Fig. 4.

Feature agglomeration using FAMD

To simplify the dataset and improve the classifier's ability to handle different data types, Factor Analysis of Mixed Data (FAMD) is used, it calculates a set of independent latent factors that capture the underlying structure and relationships in the data, allowing for the simultaneous analysis of continuous and categorical variables. The dimensionality reduction process can be formalized as Eq. (5):

$$Let\ X = \{X1,\ X2, \ldots, Xn\} \tag{5}$$

be the original set of continuous and categorical features in the dataset. FAMD computes a new set of features as in Eq. (6):

$$Y\ =\ \{Y1,\ Y2, \ldots, Ym\} \tag{6}$$

Where $m < n$, such that the variance of Y is maximized.

Modified Naïve bayes algorithm (MNB) with additive Laplace smoothing

Once the reduced set of features is obtained, the researchers apply the modified Naïve Bayes algorithm with Laplace smoothing to classify the instances as benign or botnet. The Naïve Bayes (MNB) algorithm calculates the posterior probabilities of each class given the feature values and assigns the class with the highest probability to the instance, according to Bayes' theorem in Eq. (7):

$$(C|X1,\ X2, \ldots, Xm)\ = (C)\ *\ (X1,\ X2, \ldots, Xm|C) \\ /P\,(X1,\ X2, \ldots, Xm) \tag{7}$$

Assuming conditional independence among features given the class label in Eq. (8):

$$(X1, X2, \ldots, Xm|C) = (X1|C) * (X2|C) \\ * \ldots * P(Xm|C) \tag{8}$$

To incorporate Laplace smoothing, by modifying the probability estimates as follows in Eq. (9):

$$(Xi|C)\ =\ (N\,(Xi,\ C)\ +\ \alpha)\ /\ (N\,(C)\ +\ \alpha\ *\ |V|) \tag{9}$$

where N(Xi, C) is the count of instances with feature Xi and class label C, N(C) is the total count

of instances with class label C, |V| is the number of distinct values of feature Xi, and α is the smoothing constant (typically 1). In this work, to estimate smoothed probability used the modified additive smoothed probability equation as in Eq. (10):

$$(Xi|C) = (N(Xi, C) + \alpha) / \ (N(C) + min(\alpha, Xi/|V|)) \quad (10)$$

M3L with additive Laplace smoothing

Alternatively, employing the M3L formulation to train a max-margin multi-label classifier. This approach incorporates additive Laplace smoothing and leverages previous knowledge of tightly correlated labels. After obtaining the reduced set of attributes, the M3L method is applied with Laplace smoothing to categorize the instances as either benign or botnet. The M3L algorithm computes the posterior probabilities of each class based on the feature values and assigns the instance to the class with the highest probability. The exponential number of restrictions may be decreased to a linear number while simultaneously achieving extended 1-vs.-all multi-label classification.

Finally, the results of the two proposed classifiers will be tested, and the highest classification rate will be selected for botnet detection and classification.

Results

Assessing the performance of the proposed modified Naïve Bayes and M3L algorithms for botnet detection using a dataset of network traffic containing both normal and botnet attacks instances. The dataset contains 48,000 instances, as 70% of the instances being benign and 30% are not. The researchers randomly divided the dataset into a training set contains 33,600 instances and a testing set of 14,400 instances, making sure that two sets have a balanced partitioning of normal and botnet attack instances. Fig. 5 shows the performance of the proposed combination of the modified Naïve Bayes and M3L.

Performance metrics

Accuracy as well as Mean Squared Error (MSE) have been used as a performance metrics to assess the strength of the algorithm. Accuracy measures the proportion of correct classified instances, and MSE measures the average squared difference among the predicted and actual class labels.

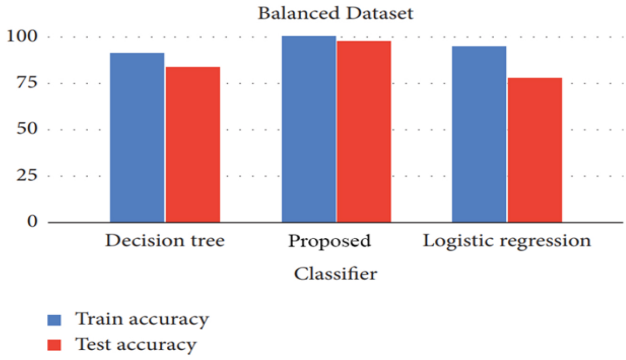

Fig. 5. Performance of the proposed modified Naïve Bayes and M3L algorithms.

To assess how well of the modified Naïve Bayes and M3L algorithms, a dataset of network traffic chose so that containing both benign and botnet instances. To ensure that both sets have a balanced spreading of normal and botnet records, the dataset is divided into training and testing sets. The suggested methods are trained on the training set and tested on the testing set. The performance metrics used to assess the effectiveness of the algorithms are accuracy and mean squared error (MSE) as in Eq. (11) and Eq. (12):

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \quad (11)$$

$$MSE = (1/N) * \ (yi - \hat{y}i)^2 \quad (12)$$

where TP, TN, FP, and FN are true positives, true negatives, false positives, and false negatives, respectively, N is the number of records in the testing set, yi is the actual class label of instance i, and ŷi is the predicted class label of instance i. The modified Naïve Bayes and M3L algorithms shown in this paper, incorporating Laplace smoothing and FAMD-based feature agglomeration, is extensively tested and compared to traditional Naïve Bayes and M3L classifiers and other state-of-the-art ML techniques. The experimental results shows that the proposed methods achieve a notable accuracy of 98.45% and a mean squared error (MSE) of 0.0390, indicating its effectiveness in accurately identifying botnet activities and mitigating the associated security risks.

Discussion

The performance results of the modified Naïve Bayes and M3L algorithms and other state-of-the-art machine learning algorithms for botnet detection shown in Table 2.

Table 2. Performance results.

Algorithm	Accuracy	MSE
Naïve Bayes	90.00%	0.207
Decision Tree	92.00%	0.159
Random Forest	92.50%	0.141
Support Vector Machine	93.50%	0.165
FADM + MNB	94.10%	0.093
FADM + M3L	95.20%	0.068
FADM + MNB + M3L	97.45%	0.039

Table 3. Top 10 most important features for botnet detection, ranked by their importance scores.

Rank	Feature	Importance Score
1	Average Packet Size	0.21
2	Flow Duration	0.17
3	Packet Length Std	0.14
4	Packet Length Mean	0.10
5	Packet Length Variance	0.09
6	Protocol	0.07
7	Destination Port	0.05
8	Source Port	0.04
9	Packet Length Max	0.03
10	Packet Length Min	0.02

The combination between MNB and M3L algorithms was achieved the highest accuracy of 97.45% and the lowest MSE of 0.039, While the MNB and M3L alone have also outperforming traditional Naïve Bayes classifiers and other state-of-the-art machine learning algorithms. These results confirm the effectiveness of the proposed methods in accurately identifying botnet activities and mitigating the associated security risks. Also, the contribution of each feature is analyzed to the classification performance using the feature importance scores provided by the Random Forest algorithm. The top ten important features for botnet detection shown in Table 3, are related to packet size, flow duration, packet length, protocol, and destination/source port. These findings suggest that these features are essential for accurately identifying botnet activities in network traffic.

Conclusion

Employing FAMD make both continuous and categorical features aggregated, enhancing the algorithm's ability to handle mixed data types and reducing the dimensionality of the feature space. This, in turn, contributes to better classification performance and reduced computational complexity. The proposed methods were evaluated using a dataset of network traffic containing both benign and botnet instances. The experimental results demonstrate that the combination of modified Naïve Bayes and M3L algorithms achieve a remarkable accuracy of 97.45% and a mean squared error (MSE) of 0.039, outperforming traditional Naïve Bayes classifiers and other state-of-the-art machine learning techniques this study highlights the potential of the combination of the modified Naïve Bayes and M3L algorithms with Laplace smoothing and FAMD-based feature agglomeration as a powerful and efficient tool for botnet detection in modern computer networks. The proposed methods offer significant improvements in accuracy and generalization capabilities, paving the way for more robust and scalable cybersecurity solutions.

Authors' declaration

- Conflicts of Interest: None.
- We hereby confirm that all the Figures and Tables in the manuscript are ours. Furthermore, any Figures and images, that are not ours, have been included with the necessary permission for republication, which is attached to the manuscript.
- No animal studies are present in the manuscript.
- No Human studies are present in the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee at Mustansiriyah University.

Authors' contribution statement

A.L.A. & B.M.N: performed conceptualization, methodology, design and implementation, also writing the original draft of this manuscript, as well as implementing the experiments and analysis of the results. A.H.A. & B.M.N.: supervised this research in terms of design, revision and proofreading.

References

1. Ibrahim MF, Alhakeem MA, Fadhil NA. Evaluation of Naïve Bayes Classification in Arabic Short Text Classification. Al-Mustansiriyah J Sci. 2021;32(4):42–50. http://dx.doi.org/10.23851/mjs.v32i4.994.
2. Haqi Al-Tai M, Nema BM, Al-Sherbaz A. Deep Learning for fake news detection: Literature review. Al-Mustansiriyah J Sci. 2023;34(2):70–81. http://dx.doi.org/10.23851/mjs.v34i2.1292.
3. Salman S, Soud JH. Deep learning machine using hierarchical cluster features. Al-Mustansiriyah J Sci. 2019;29(3):82–93. http://dx.doi.org/10.23851/mjs.v29i3.625.
4. Koroniotis N, Moustafa N, Sitnikova E. Forensics and deep learning mechanisms for botnets in Internet of Things: a survey of challenges and solutions. IEEE Access. 2019;7:61764–85. https://doi.org/10.1109/ACCESS.2019.2916717.
5. Analysis of botnet attack communication pattern behavior on computer networks. International Journal of Intelligent

Engineering and Systems (IJIES). 2022;15(4). http://dx.doi.org/10.22266/ijies2022.0831.48.
6. Padhiar S, Patel R. Behaviour based botnet detection with traffic analysis and flow intervals at the host level. Indones J Electr Eng Comput Sci. 2023;31(1):350. http://dx.doi.org/10.11591/ijeecs.v31.i1.pp350--358.
7. Le CC, Prasad PW, Alsadoon A, Pham L, Elchouemi A. Text classification: Naïve bayes classifier with sentiment Lexicon. IAENG Int J Comput Sci. 2019;46:141–8. http://dx.doi.org/10.1007/IAENG.30550129.
8. Pajila PJB, Sheena BG, Gayathri A, Aswini J, Nalini M, Subramanian S. A comprehensive survey on naive Bayes algorithm: Advantages, limitations and applications. In: 2023 4th International Conference on Smart Electronics and Communication (ICOSEC). IEEE;2023. https://doi.org/10.1109/ICOSEC58147.2023.10276274.
9. Osher S, Wang B, Yin P, Luo X, Barekat F, Pham M, *et al.* Laplacian smoothing gradient descent. Res Math Sci. 2022;9(3):4–6. http://dx.doi.org/10.1007/s40687-022-00351-1.
10. Khurana D, Koli A, Khatter K, Singh S. Natural language processing: state of the art, current trends and challenges. Multimed Tools Appl. 2023;82(3):3713–44. http://dx.doi.org/10.1007/s11042-022-13428-4.
11. Solorio-Fernández S, Carrasco-Ochoa JA, Martínez-Trinidad JF. A review of unsupervised feature selection methods. Artif Intell Rev. 2020;53(2):907–48. http://dx.doi.org/10.1007/s10462-019-09682-y.
12. Abdulhammed R, Musafer H, Alessa A, Faezipour M, Abuzneid A. Features dimensionality reduction approaches for Machine Learning based network intrusion detection. Electronics (Basel). 2019;8(3):322. http://dx.doi.org/10.3390/electronics8030322.
13. Shinan K, Alsubhi K, Alzahrani A, Ashraf MU. Machine learning-based botnet detection in software-defined network: A systematic review. Symmetry (Basel). 2021;13(5):866. http://dx.doi.org/10.3390/sym13050866.
14. Nema, Bashar M., and Shatha J. Mohammed. Secure Location Privacy Transmitting Information on Cellular Networks. Iraqi Journal of Science, Nov. 2022;63(11):5004-1, https://doi.org/10.24996/ijs.2022.63.11.35.
15. Mohammed, Shatha J., and Bashar M. Nema. Threat Detection Based on Explainable AI (XAI) and Hybrid Learning. Mesopotamian Journal of CyberSecurity, June 2025; 5(2):477–90, https://doi.org/10.58496/MJCS/2025/029.
16. Alshwely MK, AlSaad SN. Image splicing detection based on noise level approach. Al-Mustansiriyah J Sci. 2020;31(4):55–61. http://dx.doi.org/10.23851/mjs.v31i4.899.
17. Bandara AY, Weerasooriya DK, Conley SP, Bradley CA, Allen TW, Esker PD. Modeling the relationship between estimated fungicide use and disease-associated yield losses of soybean in the United States I: Foliar fungicides vs foliar diseases. PLoS One. 2020;15(6):e0234390. http://dx.doi.org/10.1371/journal.pone.0234390.
18. Abdulhameed AA, Al-Azawi RJ, Al-Mahdawi BM. Modeling web security analysis attacks with CySeMoL tool. Al-Mustansiriyah J Sci. 2020;31(3):101–9. http://dx.doi.org/10.23851/mjs.v31i3.876.
19. Hariharan B, Vishwanathan SVN, Varma M. Efficient max-margin multi-label classification with applications to zero-shot learning. Mach Learn. 2012;88(1–2):127–55. http://dx.doi.org/10.1007/s10994-012-5291-x.
20. Hariharan B, Zelnik-Manor L, Vishwanathan SVN, Varma M. Large scale max-margin multi-label classification with priors. In: Proceedings of the International Conference on Machine Learning. 2010.
21. Singh M, Singh M, Kaur S. Issues and challenges in DNS based botnet detection: A survey. Comput Secur. 2019;86:28–52. http://dx.doi.org/10.1016/j.cose.2019.05.019.
22. Kim J, Shim M, Hong S, Shin Y, Choi E. Intelligent detection of IoT botnets using machine learning and deep learning. Appl Sci (Basel). 2020;10(19):7009. http://dx.doi.org/10.3390/app10197009.

الكشف المحسّن عن شبكات الحواسيب الزائفة باستخدام خوارزمية نايف بايز المعدلة بتقنية التنعيم بواسطة لابلاس وتجميع السمات بناءً على تحليل العوامل للبيانات المختلطة.

احمد لبنان الشامي [1]، بشار مكي العيساوي[1]، علاء حسين الحمامي[2]

[1] قسم علوم الحاسوب، كلية العلوم، الجامعة المستنصرية، بغداد، العراق.
[2] قسم علوم الامن السيبراني، كلية دجلة الجامعة، بغداد، العراق.

الخلاصة

أصبحت شبكات الحواسيب الزائفة مشكلة أمنية خطيرة تهدد بشكل كبير خصوصية المستخدم وأمانة. يتطلب التصدي لشبكات الحواسيب الزائفة واكتشافها بشكل فعال تقنيات تعلم الآلة المتقدمة والمتكيفة. يهدف هذا البحث لتقديم مزيج من خوارزمية نايف بايز المعدلة مع خوارزمية M3L لتحديد شبكات الحواسيب الزائفة باستخدام تحليل العوامل للبيانات المختلطة (FAMD) لتجميع السمات وتقنية التنعيم بواسطة لابلاس للتنعيم. الأسلوب المقترح يهدف إلى التغلب على عيوب الخوارزميات التقليدية لنايف بايز، وهي عرضتها للبيانات غير المرغوبة أو الضوضاء وافتراض الاستقلال بين السمات. يستخدم الخوارزمية المحدثة تقنية التنعيم بواسطة لابلاس لمعالجة مشاكل البيانات ذات التردد الصفري وزيادة قدرة الفاحص على التعميم. علاوة على ذلك، نستخدم تحليل العوامل للبيانات المختلطة لدمج السمات المستمرة والسمات الفئوية، مما يزيد من قدرة الخوارزمية على التعامل مع أنواع البيانات المختلطة ويقلل من أبعاد مساحة السمات. ينتج عن ذلك تحسين في أداء التصنيف وتقليل في التعقيد الحسابي. باستخدام مجموعة بيانات تتضمن حركة المرور في الشبكة بما في ذلك الحوادث الطبيعية وحوادث شبكات الحواسيب الزائفة، نقوم بتقييم النهج المقترح. تشير النتائج التجريبية إلى أن مزيج من الخوارزمية المعدلة لنايف بايز مع خوارزمية M3L حققت دقة مذهلة تصل إلى 97.45% مقارنة بالأساليب الحديثة الأخرى لتعلم الآلة وخوارزميات نايف بايز التقليدية، ومعدل خطأ مربع متوسط (MSE) يبلغ 0.039. تؤكد هذه النتائج على كفاءة النهج المقترح في اكتشاف نشاط شبكات الحواسيب الزائفة بدقة وتقليل التهديدات الأمنية المرتبطة بها.

الكلمات المفتاحية: M3L , تحليل العوامل للبيانات المختلطة (FAMD)، البايز الساذج (NB)، انترنت الاشياء (IoT)، تعلم الآلة (ML)، شبكة البوتات.