Iraqi Journal of Information and Communications Technology(IJICT)
Vol. 9, Issue 1, April 2026
www.ijict.edu.iq
ISSN:2222-758X
e-ISSN:2789-7362

DETECTION OF BRUTE-FORCE LOGIN ATTEMPTS USING MACHINE LEARNING

Zaigham Abbas[1], Syed Ali Hussain[2], Basheer Ahmad[3], Muhammad Usman[4], Mushaf Ali[5], Fatima Zahra Ouariach[6], Soufiane Ouariach[7]
[1,2,3,4,5] Department of Robotics and AI Shaheed Zulfikar Ali Bhutto Institute of Science and Technology Islamabad, Pakistan
[6,7] Postdoctoral Research Fellow at AI-Explain You Science, United Kingdom & Abdelmalek Essaadi University of Tetouan, Morocco
zaighamji@gmail.com[1], syedali6160@gmail.com[2], basheerahmad389@gmail.com[3]
muhammadz.usman@gmail.com[4], mushafalimeer13579@gmail.com[5]
fatimazahra.ouariach@etu.uae.ac.ma[6], soufian.ouariach@etu.uae.ac.ma[7]
Corresponding Author: **Zaigham Abbas**
Received:25/03/2026; Revised:09/04/2026; Accepted:26/04/2026
DOI:10.31987/ijict.9.1.371

***Abstract*- Identifying brute-force and dictionary-based login attempts in modern cybersecurity systems has become increasingly challenging, as advanced techniques often fail to detect large-scale intrusion attempts.The aim of this research is to determine the effectiveness of machine learning methods in identifying such attacks in an accurately and efficiently. Two classifiers SVM and GNB, are trained on authentication log data, both with and without PCA for dimensionality reduction.The experimental results indicate that SVM achieves the highest accuracy of 97.24% without PCA and 96.55% with PCA, demonstrating that SVM is robust in high-dimensional feature spaces. Conversely, GNB shows significant with PCA, with accuracy rising from 87.93% to 91.03%, highlighting the importance of feature decorrelation in probabilistic models. The key contribution of this work is the comparative study of lightweight machine learning models demonstrating that PCA improves the performance of correlation-sensitive classifiers without undermining the computational efficiency. The results provide a feasible and scalable solution to real-time intrusion detection systems.**

keywords: **Detection System, Cybersecurity, Dictionary Attack, Authentication Security and Dimensionality Reduction.**

I. INTRODUCTION

In the current digital world, cybersecurity has become a major concern for organizations of all sizes. One of the most common threats to internet-based systems is the brute-force attack, in which attackers attempt to gain unauthorized access to systems by trying different combinations of usernames and passwords [1]. Conventional security controls,such as firewalls and intrusion detection systems, often find it difficult to detect advanced or high- volume brute-force attempts in real time [2].Machine learning is a feasible solution that enables systems to automatically detect abnormal login trends, and anticipate possible attacks using historical data. BY analyzing login activity , parameters as the frequency of logins, IP addresses, and access times can be effectively used by machine learning models to distinguish between legitimate and malicious behavior [3], [4]. This approach improves not only detection accuracy but also the response time, making the system more secure overall.One of the most persistent and dangerous cybersecurity threats is the brute-force attack, in which attackers try different combinations of the usernames and passwords until they successfully gain access to the system. Attackers can now crack millions of credentials in minutes using automated tools, including GPU-accelerated cracking

ISSN:2222-758X
e-ISSN:2789-7362

platforms and cloud-based scripts.Dictionary-based brute-force attacks enhance this technique by making it more efficient and faster through the use of large wordlists containing commonly used passwords. Modern research focuses concerned on analyzing failed login attempts,unusual timing,abnormal request frequency,and unique IP addresses to identify attacks occurring within legitimate activity patterns.Machine-learning-based intrusion detection systems have improved detection capabilities compared to signature-based systems,especially when attackers use brute-force techniques designed to mimic normal network traffic. Due to increased vulnerability to brute-force ttacks caused by the growth of cloud serices nd remote authentication systems,detection has become an important component of cybersecurity infrastructure[5-8].The fact an attacker can use brute-force attacks to generate multiple access requests using different credential combinations enable them to gain access to a system as demonstrated in Fig. 1. Large-scale and complex brute-force attacks are frequently difficult

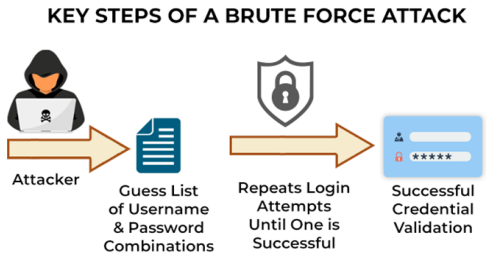

Figure 1: How the Brute force attack works [9].

for traditional security measures like firewalls and signature-based intrusion detection systems to identify, particularly when attackers imitate genuine user behavior. This drawback emphasizes the need for more sophisticated and flexible detection techniques. By examining authentication patterns and spotting anomalies in real time, machine learning offers a viable approach that enhances system security and detection accuracy.Recent work has investigated machine learning methods for identifying brute-force and authentication-based attacks, each focusing on different aspects of the problem.

As an illustration, in [1] used machine learning classifiers and achieved high detection accuracy but did not examine the effect of dimensionality reduction methods,such as Principal Component Analysis (PCA). In [2] conducted a detailed review of machine learning and deep learning methods in the field of intrusion detection, emphasizing the role of feature engineering, but they did not experimentally test the lightweight models in the context of dimensionality reduction. Conversely, deep learning-based models such as those proposed by [3] achieve strong detection performance but are computationally intensive, making them unsuitable for real-time systems. However,these studies focus on behavioral analysis and cloud authentication vulnerability and fail to provide a systematic comparison with classical machine learning models such as Support Vector Machine (SVM) and Gaussian Naive Bayes (GNB). Thus,a research gap remains regarding the impact of dimensionality reduction on various classifiers,especially lightweight models applicable to real-time cybersecurity.

To fill this gap, the this paper provides a detailed comparative analysis of SVM and GNB in identifying brute-force and dictionary-based login attacks and also evaluates the effect of PCA on model performance. The main contributions of this work are that PCA can significantly improve the performance of GNB by eliminating feature correlation, and

ISSN:2222-758X
e-ISSN:2789-7362

that SVM remains robust in high-dimensional feature space without dimensionality reduction. The paper also provides a computationally efficient intrusion detection model and validates it using performance measures such as accuracy, precision, recall, F1-score, and ROC-AUC thus providing a practical understanding of the trade-offs between model complexity and detection performance in the real-world cybersecurity context.

To address this gap,this study presents a comparative analysis of two widely used machine learning classifiers SVM and GNB for detecting brute-force and dictionary based login attacks using authentication log data. The main contributions of this work are summarized as follows:

1) A comparative evalution of SVM and GNB for brute-force attack detection.
2) An analysis of the impact of PCA on classification accuracy and model performance.
3) Identification of the effectiveness of feature decorrelation for probabilistic model such as GNB.
4) Development of a computationally efficient framework suitable for real time intrusion detection.

The remainder of this paper is organized as follows. Section II and Section III present the SVM and GNB models,respectively. Section IV discuss the PCA technique. Section VIII discuss the experimental results. Finally, Section IX concludes the paper and outlines future research directions.

II. SUPPOERT VECTOR MACHINE

Support Vector Machine (SVM) is one of the most powerful classification models applied in intrusion detection due to its strong generalization ability and ability to operate in high-dimensional feature space. According to the Fig. 2, the SVM algorithm classifies data points into different classes by determining on optimal hyperplane. SVM forms the optimal hyperplane that maximizes the separtion between normal and malicious behavior and is therefore suitable for detecting subtle anomalies such as the brute-force login patterns. SVM particularly with RBF and polynomials kernels allow it to capture nonlinear feature typical of authentication data. SVM is more accurate, precise and robust when applied to intrusion datasets than most classical algorithms, as demonstrated in numerous studies. Its strong performance on both small and large datasets along with its robustness to high-dimensional data [9-12] ,further enhances its usefulness in cybersecurity applications.

III. GAUSSIAN NAÏVE BAYES

Gaussian Naive Bayes (GNB) is a probabilistic model based on Bayes theorem, which assumes that features are independent. Though simple, GNB is quick to compute, less memory intensive, and has a reliable classification capacity; hence, it can reasonably be used real-time intrusion detection. As most authentication datasets contain continuous behavioral data (e.g. login time, time between requests and request frequency counts), the Gaussian assumption of GNB is effective in many cases. Research has shown that GNB achieves competitive performance in detecting brute-force attempts, especially when applied with preprocessing methods such as normalization or feature reduction. It is interpretable and computatationally efficient and can thus be used in a large-scal, resource-constrained systems such as cloud gateways and IoT authentication systems[13-16].One example of a Gaussian Naive Bayes (GNB) classifier used for classifying login attempts is shown in Fig. 3.

ISSN:2222-758X
e-ISSN:2789-7362

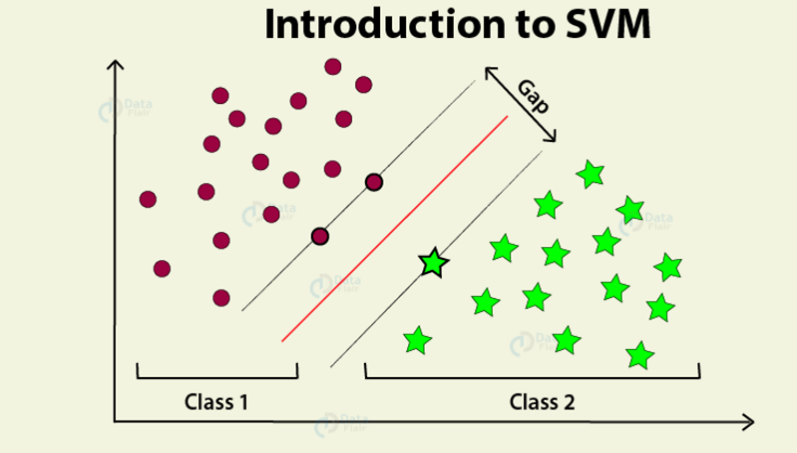

Figure 2: SVM [16].

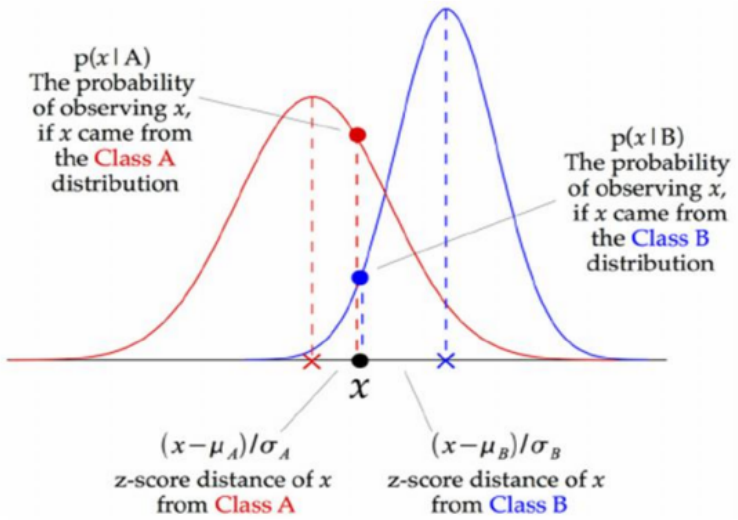

Figure 3: GNB [21].

IV. PRINCIPAL COMPONENT ANALYSIS

Principal Component Analysis (PCA) is a widely used dimensionality reduction technique for high-dimensional data, used to simplify data into a smaller set of uncorrelated components while preserving most of the variation. PCA can be applied in cybersecurity to reduce noise, remove redundant login attributes, and streamline data structures, thereby improving the effectiveness of machine-learning. Authentication datasets often contain many correlated features, e.g. failed login attempts and user behavior metrics. PCA eliminates such redundancy and improves model interpretability. It is also known to accelerate computation and improve generalization, especially in correlations-sensitive models such as GNB. PCA has been demonstrated to improve the efficiency and reliability of intrusion detection systems for event detection

ISSN:2222-758X
e-ISSN:2789-7362

[17-21]. PCA is used to reduce data dimensionality, as shown in Fig 4.

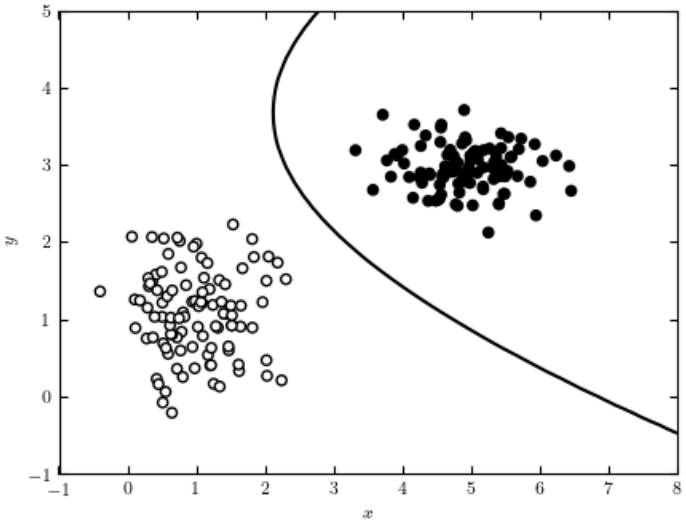

Figure 4: PCA [24].

V. DATASET

Recent research has widely investigated machine learning and deep learning methods for identifying brute-force and authentication-based attacks in cybersecurity systems due to the increasing complexity and frequency of such threats in modern network environments.

As an example, in [1] used multiple machine learning classifiers to identify brute-force attacks and achieved high accuracy; however, the authors primarily focused on classification performance, without analyzing the effect of feature reduction methods such as PCA, which is important for high-dimensional authentication data.

Other researchers such in [2], have thoroughly reviewed machine learning and transfer learning approaches to intrusion detection systems, emphasizing feature engineering and data representation, but did not conduct experimental validation of lightweight models or examine the effects of preprocessing techniques on classifier performance.

By contrast, deep learning-based frameworks, such as that proposed in [3] achieve strong detection of complex intrusion patterns but are computationally expensive, require large-scale training data,and are unsuitable for real time or resource-constrained systems such as IoT devices and cloud authentication systems. Other studies have focused on behavioral analysis methods such as login frequency of logins, IP addresses changes and access time patterns, which improve detection accuracy; however, many rely on complex architectures or lack comparative analysis of traditional machine learning methods.

Moreover, despite strong evidence that dimensionality reduction methods such as PCA to remove redundant and correlated features, improve computational efficiency, and generalization ,systematic investigation of their impact on various classifiers,

ISSN:2222-758X
e-ISSN:2789-7362

especially lightweight models such as SVM and GNB, remains limited.

Furthermore, many current studies are tested on a fixed and offline dataset, which are not representative of dynamical real-world scenarios where attack patterns constantly change, and where issues such as class imbalance, feature redundancy, scalability, and false alarm rates are not adequately addressed. Thus, a clear research gap exists in systematically comparing lightweight machine learning models in original and reduced feature spaces,as well as analyzing their suitability for real-time applications. The current research addresses these limitations by introducing a comparative framework to evaluate SVM and GNB on authentication log data with and without a PCA-based dimensionality reduction. It evaluates performance using metrics, such as accuracy, precision, recall, F1-score, and ROC-AUC, and shows that PCA significantly improves correlation-sensitive model, such as GNB, while SVM remains robust in high-dimensional features spaces.

Finally, the work provides a computationally effectient, scalable and practical solution to real-time brute-force attacks detection, thus addressing the identified research gap. The DictionaryBruteForce.csv file contains records of login an authenticate attempts. The dataset is typically provided by Klagg.com, a site known to provide sample datasets related to cybersecurity, including logs of brute-force attacks, dictionary attacks, logs of intrusion-detection testing, and indications of the similarity of an attempt to those of dictionary attacks or brute-force attacks. Examples of password patterns, frequency of attempts, time ranges, and evidence as to whether the attempt is similar to dictionary attacks or brute-force attacks are often provided. The properties enable the dataset to be applied in machine-learning classification tasks especially in discriminating between:

1) Typical efforts to log in.
2) Dictionary-based assaults.
3) Brute-force login attempts,

Such dataset has many applications in intrusion detection, cyber security studies, and development of machine learning applications to detect attacks. In table I shown dataset summary and description while table II shown the model hyper-parmeters.

TABLE I
Dataset summary and description

Attribute	**Description**
Dataset Name	Dictionary Bruteforce.csv
Number of Samples	131,477
Number of Features	135
Data Type	Mixed (Numerical + Categorical)
Source	Authentication and Network log dataset
Target Values	Login attempt Classification
Classes	Normal, Dictionary Attack, Brute-force Attack
Missing Values	Handled during preprocessing
Feature Examples	Stream ID, source MAC, Destination MAC, Packet statistics, timing featuring, IP Counts
Dimensionality Reduction	PCA applied (reduced to 5 components)

ISSN:2222-758X
e-ISSN:2789-7362

TABLE II
Model hyperparameters

Model	Parameter	Values
SVM	Kernal , Regularization(C) , Gamma, Degree	RBF ,1.0,scale,3
GNB	Variance Smoothing	$1e^{-9}$
PCA	Number of Components	5
Data Split	Training/Test Split	80% / 20%

VI. METHODOLOGY

Data Preprocessing: The data were cleaned ,missing values were handled,categorical features were encoded, and numerical features were normalized.

Dimensionality Reduction: PCA was applied to reduce feature space,and model performance with and without PCA was compared.

Model Training: Two classifiers SVM and GNB were trained on the both original data and PCA-transformed data.

Model Evaluation: Model performance was evaluated using accuracy and others performance acrossall experimental variations.

Input: Authentication Dataset D.

Output: Results of Classification (Normal / Attack).

Performance Comparison: The impact of PCA on classification accuracy and error patterns was compared for SVM and GNB models.

The proposed workflow is summarized in an Algorithm to clearly illustrate the step-by-step implementation of the detection framework.

Algorithm 1 Proposed Detection Framework

1: Load dataset D.
2: Perform data preprocessing:
- handle missing values.
- encode categorical features.
- normalize numerical features.

3: Split dataset into training and testing sets (80 % / 20 %)
4: Apply PCA to training data to obtain transformed dataset D PCA
5: Train models:
- Train SVM on the original data
- Train SVM on PCA-transformed data
- Train GNB on the original data
- Train GNB on the PCA-transformed data

6: Test all models on test dataset.
7: Evaluate performance using: accuracy, precision, recall, F1-score, ROC-AUC.
8: Compare results and analyze impact of PCA.
9: Select the Model with the best performance.

ISSN:2222-758X
e-ISSN:2789-7362

To ensure reproducibility and thorough evaluation, the experimental setup and performance metrics are clearly defined. Each experiment was conducted on a standard computing system using Python-based libraries such as Scikit-learn, Numpay, and pandas. The dataset was split into traning and testing sets using an 80/20 ratio to evaluate generalization.Multiple performance metrics,including accuracy, precision, recall, F1-score, and ROC-AUC,were used to provide a comprehensive assessment of classification effectiveness, paerticularly for imbalanced any security sensitive data.

VII. TRAIN MODELS

In this section, the following models are trained:

- Support Vector Machine (SVM) with and without PCA
- Gaussian Naive Bayes (GNB) with and without PCA
- Compare Models: Accuracy and confusion matrices are computed of each of the four models.
- Compare Results: The results are compared to analyze the effects of PCA on each classifier.

In order to provide a clear overview of the training process, the features used in this study are clearly defined and categorized based on authentication log characteristics as shown in Table III. The data includes both network-level and behavioral attributes such the frequency of login attempts, time intervals between consecutive login attempts, the numbers of IP addresses, source and destination identifiers, and packet-level metrics. These features are selected because they effectively detect abnormal patterns related to brute-force and dictionary-based attacks. For examples, a high frequency of failed logins attempts over a short period and multiple requests from a single IP address, which are strong indicators of malicious behavior. In addition, feature preprocessing techniques were applied, including the encoding of categorical variables and normalization of numerical attributes, to ensure compatiblity with machine learning models.Since the dataset contains high-dimensional and potentially correlated features, PCA was used to transform the original features into a smaller set of uncorrelated components. This not only enhance computational efficiency but also improves the performance of models that are sensitive to feature dependencies,such as Gaussian Naive Bayes. The selected features provide a meaningful representation of user behavior and network activity that facilitates effective differentiation between normal and malicious scenarios. The training phase aimed to develop classification models using both the original and PCA-transformed features. This was achieved using two machine learning algorithms: Support Vector Machine SVM and GNB.

An 80/20 split of the dataset was used to create training and testing sets. During training, each model was trained separately on:

1) The Original feature space
2) The PCA-reduced feature space

For SVM, the Radial Basis Function (RBF) kernel was applied with set hyperparameters. For GNB, the model was trained based of Gaussian distribution with variance smoothing. At this stage, no evaluation, or comparison was performed as the objective was to learn patterns from the training data.

ISSN:2222-758X
e-ISSN:2789-7362

TABLE III
Key feature used for training

Feature category	Example features	Purpose
Temporal	Login time, time interval	Detect rapid attempts
Behavioral	Failed attempts count	Identify brute-force patterns
Network	IP count,MAC address	Detect suspicious sources
Statistical	Packet statistics	Capture anomalies

VIII. RESULTS

The evaluation stage is used to assess the performance of all trained models on unseen test data. At this stage, the predictions of the models are compared using various evaluation metrics, such as accuracy, precision, recall, F1-score, and ROC-AUC. A comparative study is conducted on:

- SVM with PCA and without PCA.
- GNB with PCA and without PCA.

In addition, confusion matrices and ROC curves are analyzed to understand the classification behavior, including false positives and false negatives. This separation ensures a clear distinction between the learning process and the performance evaluation ,thereby enhancing the overall clarity and organization of the study. To ensure data integrity, the dataset was loaded and row containing missing values were removed. The attributes were separated into target labels (y) and predictors (X). The performance of the models to unseen data was evaluated using an 80-20 train-test split .

A. Classifier Models

The identification of brute-force login attempts was performed using the following four classifiers:

1) SVM without PCA: In this approach, the SVM model is trained on the original high-dimensional feature set without PCA. The model achieves an accuracy of 0.9724%, an F1-score of 0.9721%, precision of 0.9725%, recall of 0.9724%, and ROC-AUC of 0.9876%. as shown Fig. 5 ,representing the ROC for SVM without PCA. These findings suggest that SVM is high-dimensional and potentially correlated data,as it can form complex decision boundaries with strong generalization capability. This configuration provides slightly higher performance than the PCA-based SVM model, indicating that dimensionality reduction is not a mandatory requirement of SVM in this case.

2) SVM with PCA: In this approach, the SVM model is trained using features transformed by PCA,where the dimensionality is reduced to five principal components. The model achieves an accuracy of 0.9655%, an F1-score of 0.9652%, a precision of 0.9656% ,a recall of 0.9655%, and ROC-AUC of 0.9832%. These findings indicate that PCA retains most of the important information while reducing the complexity of the feature space, which consequently leads to improved computational efficiency. The performance difference between the model with and without PCA is minimal. This analysis suggests that SVM is robust to performance degreadation following dimensionality reduction, making it suitable for resource-limited environments(as shown in Fig. 6 representing the ROC curve for SVM with PCA).

ISSN:2222-758X
e-ISSN:2789-7362

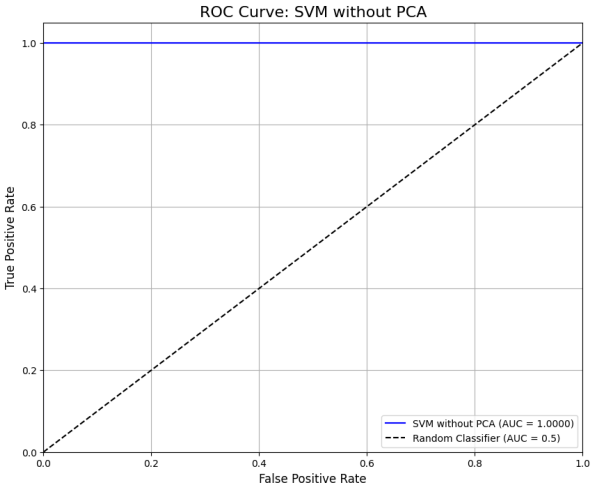

Figure 5: ROC Curve SVM without PCA.

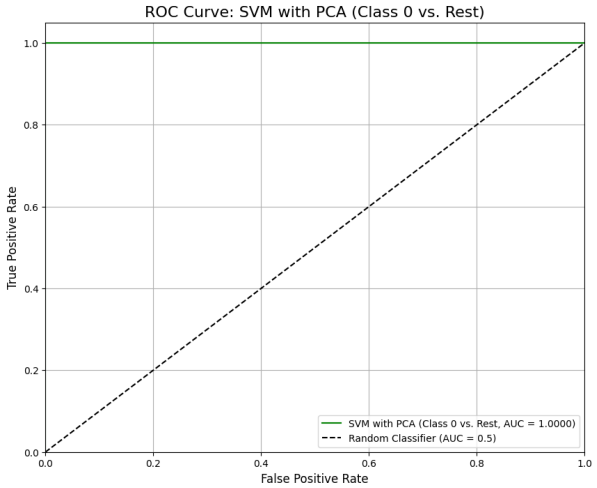

Figure 6: ROC Curve SVM with PCA.

3) Gaussian Naive Bayes (GNB) No PCA: In this instance ,the GNB model is trained on the orginal feature set without PCA. The model achieves an accuracy of 0.8793%, F1-score of 0.8790%, precision of 0.8800%, recall of 0.8793%, and ROC-AUC of 0.9135%. These findings indicate relatively lower performance compared to SVM, primarily because the independence assumption of GNB is affected by corelated features in high-dimensional data. Although the model is computationally efficient ,it is unable to capture complex relationships with the data. It also exhibits lower performance than its PCA-based counterpart, which highlights the adverse effect of feature correlation on probabilistic models (as shown in Fig. 7).

4) GNB with PCA: Here GNB model is trained on PCA-reduced features of five principal components. The model achieves an accuracy of 0.9103, F1-score of 0.9100, precision of 0.9110, recall of 0.9103, and ROC-AUC of 0.9421. These findings indicate that the results are greatly improved compared to GNB without PCA, indicating that dimensionality reduction is effective in eliminating feature correlations. PCA converts the data into a group of uncorrelated elements which is in line with the independence assumption of GNB, thus , improving its predictive power. This model outperforms the non-PCA version in all of the evaluation metrics, demonstrating the significance of preprocessing with probabilistic classifiers (shown in Fig. 8). It also demonstrates that GNB using PCA has a trade-off between the true positive rate and false positive rate at various decision thresholds. Fig 8 is the ROC curves of GNB using PCA which shows that GNB using PCA has a trade-off between the true positive rate and false positive rate at various decision thresholds. Table IV indicates the GNB with PCA consistently outperforms GNB without PCA across all evaluation metrics. Due to the capability of SVM to operate with and without PCA, the findings also give an understanding of the power of SVM to dimensionality reduction. Nevertheless, PCA over does much more feature decorrelation than GNB and it shows that feature. decorrelation is a better probabilistic modeler. Table V displays the accuracy. The results are the evidence of the dimensionality reduction resistance of SVM in that it works regardless of the PCA. However, PCA is an improvement of the GNB and it demonstrates the advantages of the decorrelation of the features to the probabilistic model. It should be mentioned that the fixed values

ISSN:2222-758X
e-ISSN:2789-7362

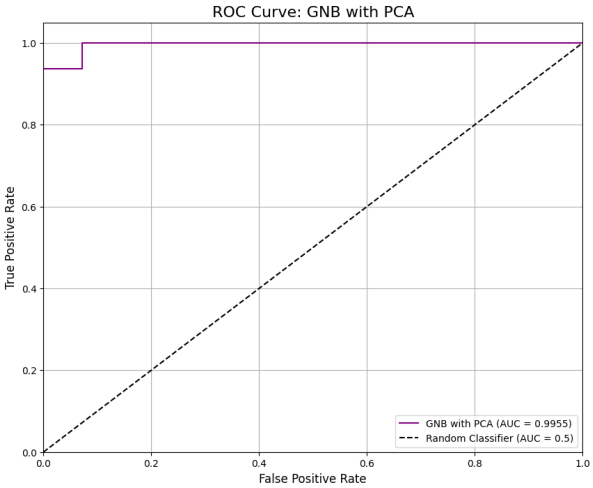

Figure 7: ROC Curve GNB with PCA.

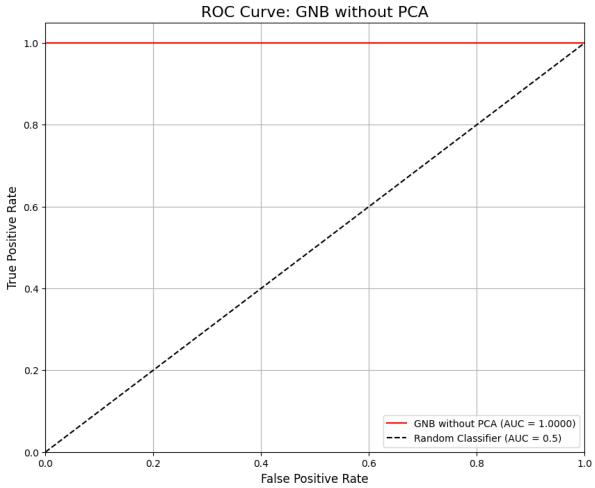

Figure 8: ROC Curve GNB without PCA.

TABLE IV
Weighted performance metrics for SVM and GNB classifiers with and without PCA

Model	F1-Score	Recall	Precision
SVM without PCA	0.9721	0.9724	0.9725
SVM with PCA	0.9652	0.9655	0.9656
GNB with PCA	0.9100	0.9103	0.9119
GNB without PCA	0.8790	0.8793	0.8800

validate the fact that GNB with PCA is better than GNB without PCA, which is discussed in the results Section. Despite the high accuracy of the SVM model being 97.24, the rest of the error value of about 2.76 is to be scrutinized, especially when cybersecurity applications are in play. False negatives (incorrectly labeled normal) of intrusion detection systems are very dangerous since they can enable attackers to access the system unauthorized. Even the slightest percentage of such errors may result in severe security breaches in the real world. False positives, conversely, can be an inconvenience to valid users, blocking access, but are not as serious as false negatives in a high-security system. Hence, reduction of false negatives needs to be given priority rather than marginal gains in overall performance. Although the study's results are encouraging, there are a number of limitations. Firstly, the models were tested in an offline environment, which does not reflect real-time system limitations like latency and streaming data processing; secondly, only a small amount of hyperparameter tuning was done, which could limit performance; and thirdly, the dataset used may not fully represent the complexity and dynamism of the real-world authentication setting, including changing patterns of attacks and distributed attack origins. The relative study shows the difference in performance of SVM and GNB models with and without PCA. We note that SVM is the most accurate when it is not used with PCA and it is robust when using high dimensional feature spaces.

TABLE V
Accuracy comparison of classification models with and without PCA

Model	Accuracy
SVM without PCA	0.9724
SVM with PCA	0.9655
GNB without PCA	0.8793
GNB with PCA	0.9103

IX. CONCLUSION

The experimental findings indicate that machine learning models can be used to differentiate between normal and malicious login attempts and that Support Vector Machine (SVM) has the highest accuracy compared to all the tested models. The high effectiveness of SVM is explained by the fact that this algorithm effectively handles high dimensional feature spaces and complex and non-linear dependencies of authentication data. Compared to probabilistic models, SVM does not assume that features are independent of each other, and thus it is less susceptible to the correlated features that are prevalent in the case of the login behavior datasets. By comparison, Gaussian Naive Bayes (GNB) performed relatively poorly on the original data because it heavily assumes that features are independent of each other. Highly correlated attributes like frequency of login, time interval, and IP pattern are normally found in authentication data and have detrimental impacts on GNB performance. Nevertheless, with the use of Principal Component Analysis (PCA), GNB accuracy was greatly improved. This is largely due to the fact that PCA converts correlated features into a set of uncorrelated features, which are more consistent with the assumptions of GNB and less noise and redundancy is present in the data. The minor drop in the SVM results when PCA is used shows that dimensionality reduction can eliminate some discriminative data that can be used to determine optimal decision boundaries. However, the feature space is reduced, resulting in better computational efficiency, which is advantageous in intrusion detection systems in real-time. In general, the results indicate that the model performance highly depends on the nature of the algorithm and dataset features. Although SVM is naturally resistant to high-dimensional and correlated data, GNB can be greatly enhanced by using preprocessing methods, such as PCA. These lessons underscore the necessity of picking proper preprocessing techniques with regard to the assumptions that the selected classifier makes. Future research could consider the more advanced models like ensemble learning and deep learning methods, and real-time deployment cases in streaming data.

FUNDING

None.

ACKNOWLEDGEMENT

The author would like to thank the reviewers for their valuable contribution in the publication of this paper.

CONFLICTS OF INTEREST

The author declares no conflict of interest.

REFERENCES

[1] A. Ali Hamza and R. Jumma surayh Al-Janabi, "Detecting brute force attacks using machine learning," BIO Web of Conferences, vol. 97, p. 00045, 2024. doi:10.1051/bioconf/20249700045.

[2] B. Hade Variant Wahono, Asfihani, I. Mahfud, B. Y. Exshadi, and A. M. Shiddiqi, "Brute Force Detection System based on machine learning classifier algorithm in cloud-based infrastructure," 2024 ASU International Conference in Emerging Technologies for Sustainability and Intelligent Systems (ICETSIS), pp. 939–943, Jan. 2024. doi:10.1109/icetsis61505.2024.10459370.
[3] K. Noor, A. L. Imoize, C.-T. Li, and C.-Y. Weng, "A review of Machine Learning and transfer learning strategies for intrusion detection systems in 5G and beyond," Mathematics, vol. 13, no. 7, p. 1088, Mar. 2025. doi:10.3390/math13071088.
[4] R. Almuhanna and S. Dardouri, "A deep learning/machine learning approach for anomaly based network intrusion detection," Frontiers in Artificial Intelligence, vol. 8, Sep. 2025. doi:10.3389/frai.2025.1625891.
[5] R. Trifonov, D. Gotseva, and P. Stoynov, "Brute Force Network Attack Detection Through Neural Networks," 2021 XXX International Scientific Conference Electronics (ET), pp. 1–4, Sep. 2021. doi:10.1109/et52713.2021.9579905.
[6] C. Shelke et al., "Machine learning-based Multi-Layer security network authentication system for uncertain attack in the Wireless Communication System," Advances in Computational Intelligence and Robotics, pp. 117–130, Dec. 2024. doi:10.4018/979-8-3373-1032-9.ch006.
[7] A. Khan and I. Sharma, "Enhancing FTP security through Ensemble Learning-based brute force attack detection," 2023 3rd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), pp. 1345–1350, Dec. 2023. doi:10.1109/icimia60377.2023.10425917.
[8] R. Dilworth, "Cloud computing and security: An overview of vulnerabilities, cyber attacks, and Ai-Driven Solutions," Proceedings of the 2024 7th Artificial Intelligence and Cloud Computing Conference, pp. 615–626, Dec. 2024. doi:10.1145/3719384.3719473.
[9] A. Doherey, A. Singh, and A. Kumar, "Intrusion detection using dense neural network in network system," 2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom), pp. 484–488, Jun. 2022. doi:10.1109/cyberneticscom55287.2022.9865436.
[10] D. Mustafa Abdullah and A. Mohsin Abdulazeez, "Machine learning applications based on SVM classification a Review," Qubahan Academic Journal, vol. 1, no. 2, pp. 81–90, Apr. 2021. doi:10.48161/qaj.v1n2a50.
[11] Y.-T. Chen and N. Ahmad, "Colorectal polyp detection and comparative evaluation based on Deep Learning Approaches," IEEE Access, vol. 11, pp. 135074–135089, 2023. doi:10.1109/access.2023.3337031.
[12] Information Technology. security techniques. Security Assurance Framework. doi:10.3403/pdisoiectr15443.
[13] S. S. Kadam et al., "Harnessing machine learning approaches for accurate energy demand forecasting in the power sector," 2024 2nd International Conference on Networking, Embedded and Wireless Systems (ICNEWS), pp. 1–6, Aug. 2024. doi:10.1109/icnews60873.2024.10731096.
[14] L. A. Tien and P. Van Huong, "Optimizing transformer models for prompt jailbreak attack detection in AI Assistant Systems," 2024 1st International Conference On Cryptography And Information Security (VCRIS), pp. 1–4, Dec. 2024. doi:10.1109/vcris63677.2024.10813380.
[15] D. Pamungkas and N. Dharmayani, "Finger movements recognition using naive Bayes algorithm in frequency domain," Proceedings of the 5th International Conference on Applied Science and Technology on Engineering Science, pp. 777–780, 2022. doi:10.5220/0011880100003575.
[16] M. A. Bouke and A. Abdullah, "An empirical study of pattern leakage impact during data preprocessing on machine learning-based intrusion detection models reliability," Expert Systems with Applications, vol. 230, p. 120715, Nov. 2023. doi:10.1016/j.eswa.2023.120715.
[17] H. Padmanaban, Comparative analysis of Naive Bayes and tree augmented naive Bayes models. doi:10.31979/etd.n7jg-e3uh.
[18] S. S A, "Comparative study of Naive Bayes, gaussian naive Bayes classifier and decision tree algorithms for prediction of heart diseases," International Journal for Research in Applied Science and Engineering Technology, vol. 9, no. 3, pp. 475–486, Mar. 2021. doi:10.22214/ijraset.2021.33228.
[19] A. Islam and M. M. Rashid, Cyberattack detection using unsupervised learning techniques, Apr. 2024. doi:10.21203/rs.3.rs-4328744/v1.
[20] B. Borzou, M. Bouchard, and H. R. Dajani, "Machine learning based listener classification and authentication using frequency following responses to English vowels for biometric applications," 2023 IEEE Sensors Applications Symposium (SAS), pp. 01–06, Jul. 2023. doi:10.1109/sas58821.2023.10254057.
[21] H. Ghani, S. Salekzamankhani, and B. Virdee, "Critical analysis of 5G networks' traffic intrusion using PCA, T-Sne, and UMAP Visualization and classifying attacks," Lecture Notes in Networks and Systems, pp. 421–437, 2024. doi:10.1007/978-981-99-6544-1_32.

