APT Attack Detection Using Packet Flow and Optimized Ensemble Machine Learning with Low Time Complexity

Kalaivani Selvaraj
School of Computer Sciences, Universiti Sains Malaysia,
Penang, Malaysia
kalaivani.selvaraj@student.usm.my

Manmeet Mahinderjit Singh
School of Computer Sciences, Universiti Sains Malaysia,
Penang, Malaysia
manmeet@usm.my

Abstract— Advanced Persistent Threat (APT) attacks steal sensitive data from targeted organizations and remain undetected by the Security Operation Center (SOC). Researchers develop automated detection of APT attacks. For efficient detection of APT attacks need, extensive data and high resources. Moreover, Deep Learning algorithms need huge labelled variables for APT attack detection. This paper uses data flow parameters such as Flow Packets, Flow IAT (Inter Arrival Time) Mean and Fwd IAT Total for APT attack detection. The Machine learning (ML) algorithms such as SVM, LR, BNN, and Bayesian Optimized Ensemble learning model are Bayesian optimized and used for APT attack detection. This study examines each ML algorithm's performance using accuracy, precision, recall, and F1-score metrics. The Bayesian Optimized Ensemble learning model performs better than traditional methods and has a high accuracy of about 97.24%, F1-score of 0.9845, and precision and recall of 0.985 and 0.9845, respectively.

Keywords— APT, Bagged Trees, Logistic Regression, SVM

I. INTRODUCTION

APTs are malicious attacks and cause substantial financial losses to organizations. APT attacks are different from conventional cyber threats in terms of their strategic planning, secretive infiltration methods, and persistent endeavours to retain hidden access in the targeted networks of organizations [1]. APT attack targets a particular organization. APT's primary goals are data exfiltration and espionage [2]. These attacks are difficult to identify using traditional cybersecurity defense methods and algorithms because they constantly change Tactics, Techniques, and Procedures (TTPs) [3]. ATP attackers use TTS to detect vulnerabilities, study behavioural patterns and enter the organization.

Moreover, organizations must invest in intrusion detection and prevention systems (IDPS) to prevent APT attacks [2, 4]. Major investments in IDPS are motivated primarily by the potential financial repercussions of attacks. APTs are currently one of the most serious threats to companies and governments [5]. However, these IDPS fail to detect new variants of APTs [6].

Machine Learning (ML) techniques can revolutionize cybersecurity [7-8]. These techniques, which learn independently and adjust to changing patterns in data, can identify complex patterns of unusual activity and previously learned patterns based on various datasets [9]. This learning flexibility enhances the detection of APT threats based on behavioural patterns, inspiring hope for more effective cybersecurity measures [10].

[11] classifies the APT attack detection based on the host and the network traffic. Host-based detection systems use classification models such as random forest (RF), Naïve Bayes (NB), and decision trees (DT) and analyze the network connectivity, Central Processing Unit (CPU) usage, memory access, and process creation. Network traffic-based detection collects communication traffic data, analyses the feature, and detects the attack using support vector machine (SVM) and recurrent neural networks (RNN) [12].

The stealthy nature of APT ransomware requires an effective attack detection system capable of detecting it based on features. The attack is achieved through abnormal patterns in computer networks over an extended period of time [11], which distinguishes the APT attack based on false positives and false negatives [6].

A. Problem Statement

Traditional APT attack detection focuses on network traffic analysis through ML models with many features. However, ML models consume more execution time [16, 17, 18, 19]. The traffic features used in the above ML models have a falser negative in the classification of the APT attack due to a large number of datasets. To solve the above problem and to improve accuracy, the present study detects APT attack traffic classification using machine learning models with small data and low time complexity. The proposed model uses the Bayesian Optimized Ensemble learning method that improves the overall performance of the APT detection system by minimizing space and time complexity.

B. Contributions

The contributions of this paper are as follows: (i) To detect the APT attack with fewer features such as Flow Packets, Flow IAT Mean and Fwd IAT Total using Bayesian optimized machine learning algorithms. (ii) To propose that the performance of optimized classifiers such as SVM, LR, and BNN for APT attack detection be estimated. (iii) To compare different classification techniques using various measures such as accuracy, precision, recall, and F1-score for APT detection.

This paper consists of five sections. Section 2 covers the related work of APT attack detection. Section 3 describes the methodology and experimental analysis. Section 4 discusses the results and findings. Section 5 provides the conclusion.

II. RELATED WORK

This section explains the related work of the APT attack. Limitations of prior research on APT attack detection techniques include a high percentage of false attack detection and an inability to identify assaults in real-time. APT attack detection using conventional intrusion detection systems (IDS) is challenging. Conventional systems depend on fixed signatures [13]. Researchers use ML approaches for the identification of APT attacks [14]. The Researchers use features such as user behaviour, network traffic data, and system logs to train the models for APT detection [15]. Table 1 discusses the summary of the literature review.

The Machine learning-based APT (MLAPT) method used in [6] analyses the network traffic using simulated data due to the lack of relevant data sources and predicts the APT attacks with low false positive rates. An active learning-based method discussed in [16] detects the C&C Server in the APT attack. RF is used to analyse and classify the abnormal behavioural features of the network traffic. The method fails

2024 IEEE Symposium on Wireless Technology & Applications (ISWTA) | 979-8-3503-5135-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/ISWTA62130.2024.10652055

to detect the APT attack using encryption techniques during transmission. Moreover, [17] trained a multiclass model (SMOTHE-RF) to deal with the imbalance and multiclassification problems in the dynamic analysis of APT malicious software event logs. The feature extraction could be more effective due to the samples used in the study.

A temporal learning method, used in [18], dynamically analyses the files (.exe PE) based on a temporal segment on the occurrences of API calls during the PE's execution—the method's effectiveness leverages due to low-level behavioural patterns and high time complexity. The early discovery of the APT attack approach by [19] uses the SVM method, extracts the network traffic features using principal component analysis (PCA), and enhances detection efficiency. The data dimensionality reduction method improves the efficiency of the detection system. The ontology model introduced by [20] identifies and cognizes the APT malware by extracting the dynamic system call based on behaviour characteristics. A comprehensive evaluation is performed manually on APT attack detection with a high detection time. Additionally, an automated multi-view consensus clustering technique implemented by [21] defines an optimum defense decision to identify the source of the attackers. [22] improves the effectiveness of the system by using different ML models.

TABLE 1. SUMMARY OF RELATED WORK

Proposed Model & Ref	Dataset	Platform	Features used	Classifiers
MLAPT [6]	Simulated	Network	Network Traffic	SVM
C&C Server Detection Model [16]	CTU-13	Network	Network Traffic	RF
SMOTE-RF model [17]	NSFOCUS	Windows	Event logs	KNN, DT, XGBoost
Bon-APT [18]	Simulated	Windows	Timestamped API calls	SVM
Radial Basis Function-Support Vector Machine (SVM-RBF) [19]	NSL-KDD	Network	Network traffic	SVM, J48 DT, NB, Multilayer Perceptron (MLP)
APT MalInsight [20]	Hangover, DarkHotel, Mirage, NormanShark, SinDigoo	Windows	API calls	TF-IDF, RF
Multi-view Fuzzy Consensus Clustering (MFCC) Model [21]	Simulated	Windows	Header information, Binary Opcode, Bytecode, and API calls	Fuzzy pattern tree and Multi-modal fuzzy classifier
Present study	SCVIC-APT-2021	Windows	Flow Packets, Flow IAT Mean and Fwd IAT Total	Bayesian Optimized Ensemble learning

However, APT attacks' accuracy, detection time, and feature selection remain challenging. This paper proposes the Bayesian Optimized Ensemble learning ML model for detecting APT attacks with more minor number data, and Flow features such as Flow Packet, Flow IAT Mean, and Fwd IAT Total. Hence, the Bayesian Optimized Ensemble learning model reduces computational resources and accurately detects the APT attack with a small dataset.

III. METHODOLOGY

This study addresses the limitations in the existing literature and fills in the research gap. It examined and classified network traffic flow parameters and classified the APT utilizing a small number of data and current data. The proposed model consists of Bayesian Optimized Ensemble bagged trees. Bagged trees reduce the effects of overfitting and improve stability and generalization. Fig. 1 shows the APT Attack Detection Methodology in Windows.

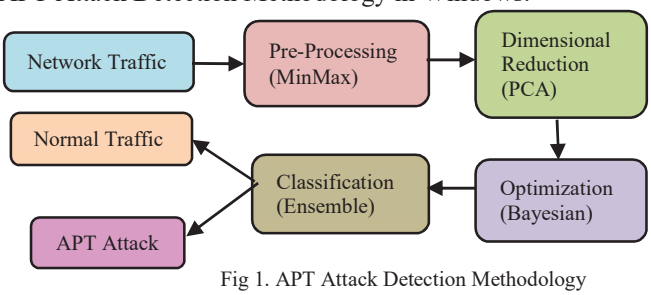


Fig 1. APT Attack Detection Methodology

This study uses a publicly available Kaggle dataset, a network traffic dataset, and the MinMax filtering algorithm for pre-processing. Next, PCA is used to reduce the dataset's dimensions and select features. A Bayesian algorithm optimizes the ML algorithms and selects the features. After choosing the features, different classification models are used to classify data for APT attack detection.

A. Dataset Description

The SCVIC-APT-2021 dataset from Kaggle has been collected and published by [23]. In this paper, to detect APT attacks in network traffic and evaluate the APT detection performance, the SCVIC-APT-2021 dataset is used. The dataset contains 56,487 records and 84 attributes (including the class attribute). Out of the 56,487 records, 55,583 are regular traffic. Additionally, the dataset contains all the values. The dataset includes five stages of APT attack and regular traffic (6 labels in total). The stages of an APT attack are as follows: Initial compromise (77 records), Reconnaissance (251 records), Pivoting (360 records), Lateral movement (142 records), and Data Exfiltration (75 records). Moreover, Table 2 shows all the features present in the dataset.

TABLE 2. FEATURES FOUND IN SCVIC-APT-2021 DATASET

#	Features	#	Features	#	Features
1	Flow ID	29	Fwd IAT Std	57	ECE Flag Count
2	Src IP	30	Fwd IAT Max	58	Down/Up Ratio
3	Src Port	31	Fwd IAT Min	59	Average Packet Size
4	Dst IP	32	Bwd IAT Total	60	Fwd Segment Size Avg
5	Dst Port	33	Bwd IAT Mean	61	Bwd Segment Size Avg
6	Protocol	34	Bwd IAT Std	62	Fwd Bytes/Bulk Avg
7	Timestamp	35	Bwd IAT	63	Fwd

			Max		Packet/Bulk Avg
8	Flow Duration	36	Bwd IAT Min	64	Fwd Bulk Rate Avg
9	Total Fwd Packet	37	Fwd PSH Flags	65	Bwd Bytes/Bulk Avg
10	Total Bwd packets	38	Bwd PSH Flags	66	Bwd Packet/Bulk Avg
11	Total Length of Fwd Packet	39	Fwd URG Flags	67	Bwd Bulk Rate Avg
12	Total Length of Bwd Packet	40	Bwd URG Flags	68	Subflow Fwd Packets
13	Fwd Packet Length Max	41	Fwd Header Length	69	Subflow Fwd Bytes
14	Fwd Packet Length Min	42	Bwd Header Length	70	Subflow Bwd Packets
15	Fwd Packet Length Mean	43	Fwd Packets/s	71	Subflow Bwd Bytes
16	Fwd Packet Length Std	44	Bwd Packets/s	72	FWD Init Win Bytes
17	Bwd Packet Length Max	45	Packet Length Min	73	Bwd Init Win Bytes
18	Bwd Packet Length Min	46	Packet Length Max	74	Fwd Act Data Pkts
19	Bwd Packet Length Mean	47	Packet Length Mean	75	Fwd Seg Size Min
20	Bwd Packet Length Std	48	Packet Length Std	76	Active Mean
21	Flow Bytes/s	49	Packet Length Variance	77	Active Std
22	Flow Packets/s	50	FIN Flag Count	78	Active Max
23	Flow IAT Mean	51	SYN Flag Count	79	Active Min
24	Flow IAT Std	52	RST Flag Count	80	Idle Mean
25	Flow IAT Max	53	PSH Flag Count	81	Idle Std
26	Flow IAT Min	54	ACK Flag Count	82	Idle Max
27	Fwd IAT Total	55	URG Flag Count	83	Idle Min
28	Fwd IAT Mean	56	CWR Flag Count	84	Label

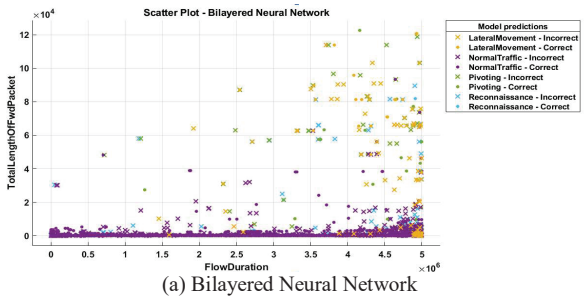
B. *Implementation Setup*

This study used the Windows-based MATLAB2024a. The system configuration is an i7 core processor with 16 GB RAM. In this study, data pre-processing uses Min-Max scaling. Dimensional feature selection uses Principal Component Analysis (PCA), and 20 out of 84 features are selected. Ensemble methods such as bagged trees reduce the bias towards the majority class. They solve the problem of imbalanced data.

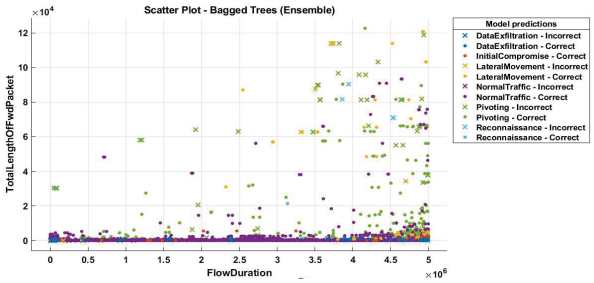
Moreover, the test uses different methods, such as SVM, LR, and BNN, with 20 features. The selected features are Flow Duration, Total Fwd Packet, Total Bwd Packet, Total Length of Fwd Packet, Total Length of Bwd Packet, Fwd Packet Length Max, Fwd Packet Length Min, Bwd Packet Length Max, Bwd Packet Length Min, Packet Length Min, Packet Length Max, Fwd Packets/s, Bwd Packets/s, Average Packet Size, Fwd Segment Size Avg, Fwd Segment Size Avg, Active Mean, Active Std, Idle Mean, Idle Std. In the proposed ML algorithms, such as SVM, LR, BNN and Bayesian Optimized Ensemble learning methods, the Hyperparameter tunes the Bayesian optimizer for APT attack detection and Variant APT attack detection.

C. *APT Attack Detection Using Bayesian Optimize ML Algorithms*

A scatter plot is an analytical technique for identifying outliers and providing a thorough knowledge of the impact. Principal Component Analysis (PCA) is used for feature selection and dimension redundancy, and a Bayesian optimizer is used to optimize the ML model's Hyperparameter.



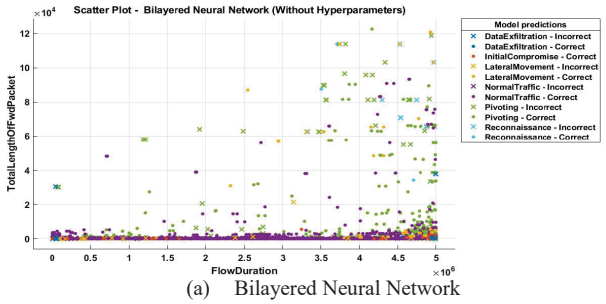
(a) Bilayered Neural Network



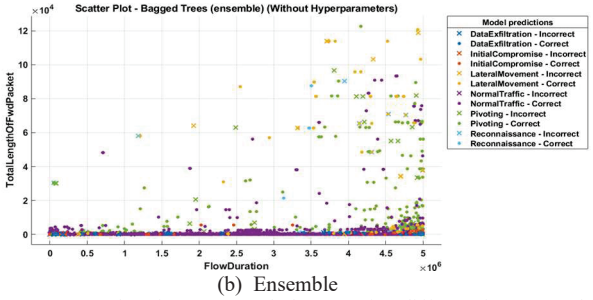
(b) Ensemble

Fig 2. Scatter Plots for APT attack dataset using different features and classifiers using Hyperparameters

Fig. 2 depicts the scatter plots for APT datasets for different stages on Bilayered Neural Networks and Ensemble classifiers using PCA and optimizer.



(a) Bilayered Neural Network

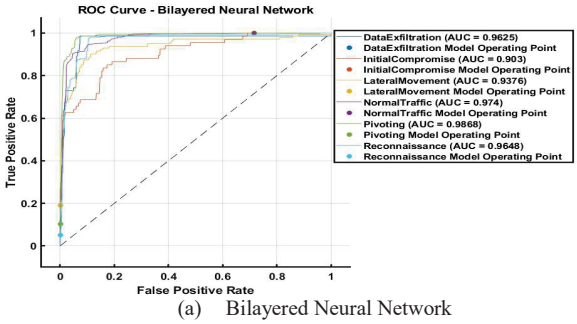


(b) Ensemble

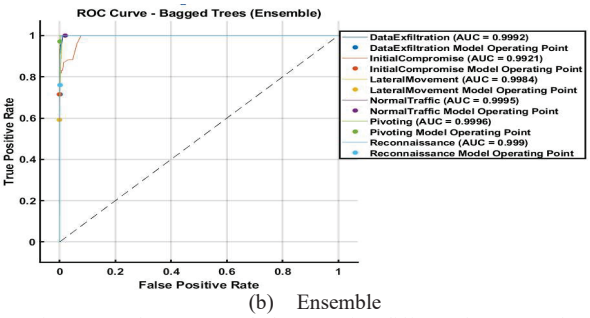
Fig 3. Scatter Plots for APT attack dataset using different features and classifiers without using Hyperparameters

D. ROC Based APT Attack Detection

The Receiver Operating Characteristic Curve (ROC) is a method for arranging, choosing, and displaying classifiers according to their performance in the (ROC) graph, as shown in Fig 4. The curve plots validate the models' True Positive (TP) and False Positive (FP) rates.



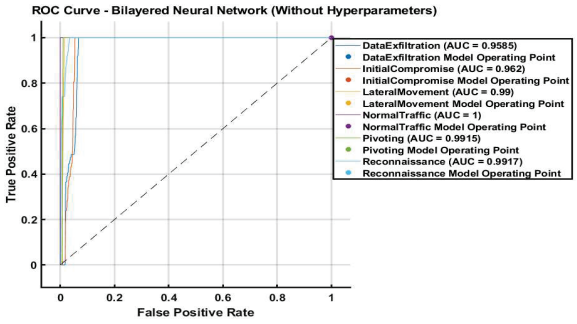
(a) Bilayered Neural Network



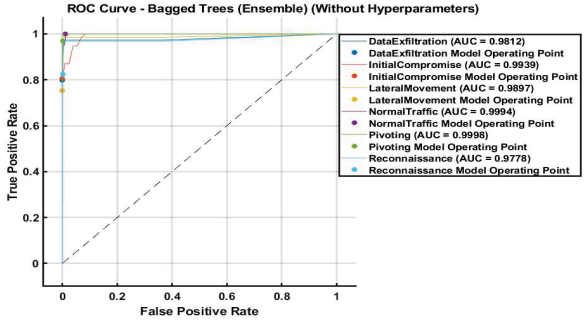
(b) Ensemble

Fig 4. ROC for APT attack dataset using different features and classifiers using Hyperparameters

Fig 5. shows the ROC without using PCA and Bayesian optimizer and evaluates the performance of the different stages on Bilayered Neural Network and Ensemble classifiers without using PCA and optimizer.



(a) Bilayered Neural Network



(b) Ensemble

Fig 5. ROC for APT attack dataset using different features and classifiers without using Hyperparameters

Figs. 4 and 5 show the TPs and FPs of different classifiers using PCA with and without a Bayesian Optimizer.

IV. RESULTS AND DISCUSSIONS

This study considers four evaluation metrics for APT attack stage detection: accuracy, recall, F1 score, and precision. The evaluation was done and recorded using these metrics. Table 2 and Fig 6 show SVM, LR, BNN, and a Proposed Model model comparison and depict performance with and without using Bayesian and PCA for 20 features.

With the Bayesian optimizer and PCA as hyperparameters, the Bayesian Optimized Ensemble learning model has achieved the highest precision value at 98.5%, followed by SVN at 90.5%, BNN at 90.49%, and LR at 90.37%, which is the lowest precision value. Bayesian Optimized Ensemble learning model achieved 97.24% as the highest accuracy, 90.44% accuracy obtained with BNN, 88.97% accuracy obtained using LR, and SVM with 89.95% as the lowest value. The performance of the model is critical and measured using a recall metric. The recall value is more excellent than 90% for all the models. SVM has achieved the highest recall of 100%, relatively LR with the lowest value of 97.74%, BNN with 99.93% and the Bayesian Optimized Ensemble learning model with 98.45%. Regarding the F1-score, the Bayesian Optimized Ensemble learning model has achieved 98.45% and performs better than traditional models. Meanwhile, the F1-score of BNN was 94.90%, SVM was 94.12%, and LR was 93.56%.

Table 3 discusses techniques used on different datasets with various features and measures their accuracy. The comparison is analyzed and concludes that the proposed Bayesian Optimized Ensemble learning method provides better 97.24% accuracy, 98.5% precision, 98.45% F1-score, and 98.45% recall than other existing methods. The recall metric is very important for performance evaluation.

True Class							
	DE			2	65	1	6
	IC			12	41	21	3
	LM			27	89	19	7
	NT			17	53729	7	3
	P			40	238	37	45
	R			8	217	13	13
		DE	IC	LM	NT	P	R
		Predicted Class					

(a) Bilayered Neural Network

True Class							
	DE	59		1		4	10
	IC		62		10	5	
	LM	3		107		8	24
	NT				53756		
	P	1	3			349	7
	R	13		16		15	207
		DE	IC	LM	NT	P	R
		Predicted Class					

(b) Ensemble

DE – DataExfiltration, IC – Initial Compromise, LM – Lateral Movement, NT – NormalTraffic, P – Pivoting, R - Reconnaissance

Fig 7. Confusion Matrix for Multiclass Function

TABLE 2. PERFORMANCE ANALYSIS BASED ON 20 FEATURES

Model	Accuracy		Precision		F1 - Score		Recall	
	With Bayesian	*Without Bayesian*	*With Bayesian*	*Without Bayesian*	*With Bayesian*	*Without Bayesian*	*With Bayesian*	*Without Bayesian*
SVM	89.95 %	89.95 %	90.5%	89.5%	94.12%	94.12%	100%	100%
LR	88.97%	88.43%	90.37%	90.10%	93.77%	93.56%	97.74%	97.68%
BNN	90.44%	90.24%	90.49%	90.17%	94.90%	94.80%	99.93%	99.91%
Proposed model (Ensemble)	97.24%	96.89%	98.50%	98.29%	98.45%	98.25%	98.45%	98.25%

Comparison of different datasets using different classifiers is analyzed, and their performance is compared and shown in Table 3.

TABLE 3. COMPARISON BETWEEN THE PROPOSED METHOD AND OTHER EXISTING METHODS

Technique used	Dataset used	# of Features used	Accuracy	Precision	F1-Score	Recall	Ref.
SVM	Simulated dataset	7 attributes	84.8 %	-	-	-	[6]
RF	CTU-13	17	99.98 %	99.96 %	-	1.00 %	[16]
KNN, DT, XGBoost	NSFOCUS	20	80 %	95.7 %	83 %	73.3 %	[17]
SVM	Simulated dataset	500 Samples	97.7 %	-	-	-	[18]
SVM, J48 DT, NB, MLP	NSL-KDD	94	96.4 %	-	-	-	[19]
RF	Hangover, DarkHotel, Mirage, NormanShark, SinDigoo	24	99.28 %	-	-	-	[20]
Fuzzy pattern tree and Multi-modal fuzzy classifier	Simulated dataset	1200 samples	95.2 %	-	-	-	[21]
Bayesian Optimized Ensemble learning	SCVIC-APT-2021	20	97.24 %	98.5 %	98.45 %	98.45 %	Our Study

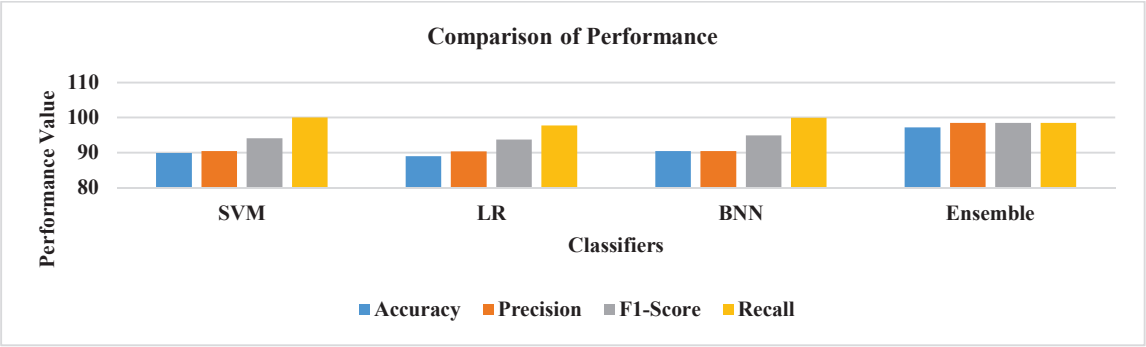


Fig 6. Comparison results on different classifiers' performance based on various metrics

Fig. 7 depicts the confusion matrix for the multiclass function for all six labels, such as Initial compromise, Reconnaissance, Pivoting, Lateral movement, Data Exfiltration, and Normal traffic data. TP represents the identification of an APT attack. For normal traffic, TN is used for identification. FP is incorrect identification of APT attack as normal traffic. FN denotes the wrong identification of normal traffic. If FPs are more than FNs, then the performance of the Bayesian Optimized Ensemble learning model with 53756 is considered more accurate.

V. CONCLUSION

Advanced Persistent Threats (APT) are network attacks that use multiple stages and different attack techniques. APT attackers plan their attack strategies based on the specific targets and perform the attack over some time. APT attacks perform in multiple stages, such as initial access, first penetration and malware deployment, expanded access and move laterally, exfiltration or damage infliction, and finally, they perform follow-up attacks. This study uses the SCVIC-APT-2021 dataset to detect APT attacks using proposed optimized ML techniques. MinMax scaling is to pre-process the data and improve its quality. PCA is for feature selection, where 20 out of 84 features are selected using PCA. The 20 features use the proposed optimized machine learning algorithms such as SVM, LR, BNN, and Bayesian Optimized Ensemble Learning Model. The performance of these models evaluates accuracy, precision, recall, and F1-score. From analysis, the ensemble model has 97.24% accuracy, 98.45% of F1-score, and the highest average recall of about 98.45% compared to traditional methods. This study focused on detecting APT attacks based on network traffic analysis of SVM with 89.95 % accuracy, low time complexity, and fewer features. The execution time of the system is fast and accurate compared to the Bayesian Optimized Ensemble learning model. In the future, researchers can detect APT attacks on devices like mobile and IoT with small datasets and reduce the organisation's risk.

FUNDING

The Ministry of Higher Education Malaysia supports this work under the Fundamental Research Grant Scheme, project Code FRGS/1/2020/ICT07/USM/02/2.

REFERENCES

[1] J. M. Rugina, "Through the Eyes of Attackers: A Comprehensive Analysis of CyberSecurity Strategies in International Relations," *Afro Eurasian Studies*, vol. 12, no. 1, pp. 40-57, 2023.

[2] T. Jabar and M. Mahinderjit Singh, "Exploration of mobile device behaviour for mitigating advanced persistent threats (APT): a systematic literature review and conceptual framework," *Sensors*, vol. 22, no. 13, pp. 4662, 2022.

[3] A. A. Al-Kadhimi, M. M. Singh, and M. N. A. Khalid, "A systematic literature review and a conceptual framework proposition for APT detection for mobile devices using artificial intelligence techniques," *Applied Sciences*, vol. 13, no. 14, p. 8056, 2023.

[4] D. T. Salim, M. M. Singh, and P. Keikhosrokiani, "A systematic literature review for APT detection and Effective Cyber Situational Awareness (ECSA) conceptual model," *Heliyon*, 2023.

[5] R. M. Rajendran and B. Vyas, "Detecting APT Using Machine Learning: Comparative Performance Analysis With Proposed Model," in *SoutheastCon 2024*, 2024, pp. 1064-1069.

[6] I. Ghafir, M. Hammoudeh, V. Prenosil, L. Han, R. Hegarty, K. Rabie, and F. J. Aparicio-Navarro, "Detection of advanced persistent threat using machine- learning correlation analysis," *Future Generation Computer Systems*, vol. 89, pp. 349-359, 2018.

[7] T. Jabar, M. M. Singh, and A. A. Al-Kadhimi, "Mobile Advanced Persistent Threat Detection Using Device Behavior (SHOVEL) Framework," in *Proc. 8th Int. Conf. Comput. Sci. Technol. (ICCST 2021)*, Labuan, Malaysia, Mar. 2022, pp. 495-513. Singapore: Springer Singapore.

[8] M. N. A. Khalid, A. A. Al-Kadhimi, and M. M. Singh, "Recent developments in game-theory approaches for the detection and defense against advanced persistent threats (APTs): a systematic review," *Mathematics*, vol. 11, no. 6, p. 1353, 2023.

[9] A. A. Al-Kadhimi, M. M. Singh, and T. Jabar, "Fingerprint for mobile-sensor apt detection framework (FORMAP) based on TTP and Mitre," in *Proc. 8th Int. Conf. Comput. Sci. Technol. (ICCST 2021)*, Labuan, Malaysia, Aug. 28-29, 2021, pp. 515-533. Singapore: Springer Singapore, Mar. 2022.

[10] G. Apruzzese, P. Laskov, E. Montes de Oca, W. Mallouli, L. Brdalo Rapa, A. V. Grammatopoulos, and F. Di Franco, "The role of machine learning in cybersecurity," *Digital Threats: Research and Practice*, vol. 4, no. 1, pp. 1-38, 2023.

[11] X. Wang, Q. Liu, Z. Pan, and G. Pang, "APT attack detection algorithm based on spatio-temporal association analysis in industrial network," *Journal of Ambient Intelligence and Humanized Computing*, pp.1- 10, 2020.

[12] Y. Chen, S. Kar, and J. M. Moura, "Dynamic attack detection in cyber-physical systems with side initial state information," *IEEE Transactions on Automatic Control*, vol. 62, no. 9, pp. 4618-4624, 2016.

[13] G. Brogi, "Real-time detection of advanced persistent threats using information flow tracking and hidden Markov models," (Doctoral dissertation, Conservatoire national des arts et metiers-CNAM), 2018.

[14] J. Gu, R. Kong, R, H. Sun, H, H. Zhuang, F. Pan, and Z. Lin, "A novel detection technique based on benign samples and one-class algorithm for malicious PDF documents containing JavaScript," *International Conference on Computer Application and Information Security (ICCAIS 2021)*, vol. 12260, pp. 599-607, SPIE, 2022.

[15] A. S. AL-Aamri, R. Abdulghafor, S. Turaev, I. Al- Shaikhli, A. Zeki, and S. Talib, "Machine Learning for APT Detection," *Sustainability*, vol. 15, no. 18, pp. 13820, 2023.

[16] C. Do Xuan, L. Van Duong, and T. V. Nikolaevich, "Detecting C&C server in the APT attack based on network traffic using machine learning," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 11, no. 5, 2020.

[17] S. Li, Q. Zhang, X. Wu, W. Han, and Z. Tian, "Attribution classification method of APT malware in IoT using machine learning techniques," *Security and Communication Networks*, pp. 1-12, 2021.

[18] G. Shenderovitz, and N. Nissim, "Bon-APT: Detection, attribution, and explainability of APT malware using temporal segmentation of API calls," *Computers & Security*, vol. 142, pp. 103862, 2024.

[19] W. L. Chu, C. J. Lin, and K. N. Chang, "Detection and classification of advanced persistent threats and attacks using the support vector machine," *Applied Sciences*, vol. 9, no. 21, pp. 4579, 2019.

[20] W. Han, J. Xue, Y. Wang, F. Zhang, and X. Gao, "APTMalInsight: Identify and cognize APT malware based on system call information and ontology knowledge framework," *Information Sciences*, vol. 546, pp. 633-664, 2021.

[21] H. Haddadpajouh, A. Azmoodeh, A. Dehghantanha, and R. M. Parizi, "MVFCC: A multi-view fuzzy consensus clustering model for malware threat attribution," *IEEE Access*, vol. 8, pp. 139188-139198, 2020.

[22] N. Jeffrey, Q. Tan, and J. R. Villar, "A review of anomaly detection strategies to detect threats to cyber- physical systems," *Electronics*, vol. 12, no. 15, pp. 3283, 2023.

[23] J. Liu, Y. Shen, M. Simsek, B. Kantarci, H. T. Mouftah, M. Bagheri, and P. Djukic, "A new realistic benchmark for advanced persistent threats in network traffic," *IEEE Networking Letters*, vol. 4, no. 3, pp. 162-166, 2022.