International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161
©2023 INPRESSCO®, All Rights Reserved
Available at http://inpressco.com/category/ijcet

Research Article

Evaluating the Effectiveness of Machine Learning (ML) Models in Detecting Malware Threats for Cybersecurity

Mani Gopalsamy[1*]

[1]Senior Cyber Security Specialist Louisville, KY, USA- 40220

Received 26 Nov 2023, Accepted 05 Dec 2023, Available online 15 Dec 2023, **Vol.13, No.6 (Nov/Dec 2023)**

Abstract

Cybersecurity is rapidly embracing ML. Integrating ML into cybersecurity mainly aims to improve the effectiveness, scalability, and actionability of malware detection compared to more conventional approaches that depend on human intervention. Problems with ML need well-managed theoretical and methodical approaches in the cybersecurity sector. The increasing prevalence of cyber threats necessitates effective strategies for malware detection within cybersecurity frameworks. Using the EMBER v2017 dataset—this study intends to develop and assess ML methods for malware attack detection and classification. This research used machine learning classification algorithms Neural Network (NN), Random Forest (RF), and SVM (Support vector machine) and evaluated the performance of these models in terms of F1 score, precision, accuracy, and recall. The Neural Network model exceeds the others, with an accuracy of 97.53% and a precision of 98.85%, whereas RF has a lesser accuracy of 84.3%. These findings underscore the importance of using powerful machine-learning techniques to improve cybersecurity safeguards against emerging threats. The work contributes to the field by providing a detailed examination of the performance of several malware detection techniques, as well as recommendations for future research and practical cybersecurity applications.

Keywords: *Malware Detection, Cybersecurity, EMBER Dataset, machine learning, classification algorithms.*

1. Introduction

In the last few decades, as information technology has gotten more and more popular, a number of security challenges have developed, including virus attacks, denial of service (DoS), unauthorised acces[1], Zero-day attacks, data breaches, social engineering, phishing, and related activities have increased exponentially during the last ten years. Malware researchers and analysts recorded less than 50 million unique executables in 2010. This claimed figure increased by around 100% in 2012, reaching over 100 million. The security industry discovered more than 900 million malicious executables in 2019, according to AV-TEST statistics, and that figure is consistently increasing [2]. People and companies alike are vulnerable to cybercrime and network assaults, which may result in substantial financial losses[3]. For instance, data breaches cost an estimated US$8.19 million worldwide and an average of US$3.9 million. On top of that, every year, the economy loses $400 billion due to cybercrime[4]

A survey of security experts found that in the next five years, the amount of compromised documents would almost triple[5]. To avoid such losses, firms must establish and implement an effective cybersecurity plan[6][7]. Recent socioeconomic studies have shown that governments and individuals having access to highly classified data, apps, and technologies are crucial to national security[8][9][10]. It is also contingent upon the businesses that provide their employees with access, as they have the ability and knowledge to promptly and effectively identify such cyber threats [11][12]. As a result, protecting vital systems against cyberattacks—both known and unknown—and intelligently identifying a wide range of cyber-occurrences are the two most important issues that need our urgent attention[13][14].

Cybersecurity is the study and practice of preventing harm to, or intrusion into, computer systems, networks, programs, and data by the use of appropriate technological measures[15][16][10]. There are numerous subfields within cybersecurity, which encompass a wide range of scenarios from enterprise to mobile computing[17][18]. First, there's network security, which is all about keeping hackers out of networks; second, there's application security, which is about making sure that software and hardware are safe from cyber threats [19][20]; third, there's information security, which is mostly about

*Corresponding author's ORCID ID: 0000-0000-0000-0000
DOI: https://doi.org/10.14741/ijcet/v.13.6.4

protecting sensitive data; and lastly, there are operational security measures to take when dealing with data assets[21][22]. Conventional approaches to computer and network protection include firewalls, antivirus software, and IDS[23][24]. One of the leading forces in this process is data science, which leverages ML, or Machine Learning, a subfield of "Artificial Intelligence" to analyse massive data for meaningful insights. Data science has evolved a new paradigm in science[25], and ML has transformed cybersecurity [26].

Firewalls, encryption, and the implementation of user ID/access control were drastic security measure forms, which in the past provided adequate security to today's cyber businesses[27]. In this respect however, machine learning can be considered as a partial but significant paradigm shift[28][29]. One of the major potential issues is that when there is a need to resolve ad hoc data management, both domain specialists and security analysts must do it independently. But as more and more cybersecurity incidents in a variety of forms surface over time, conventional approaches to controlling these cyber-hazards have shown to be ineffectual. Consequently, several intricate and novel assaults surface and swiftly proliferate throughout the network.

The goal of this research is to create and evaluate effective techniques for detecting malware threats within cybersecurity systems. By leveraging machine learning algorithms and advanced data analysis methods, the study aims to identify and classify potential malware attacks, improving the ability to prevent, mitigate, and respond to security breaches. The research will focus on enhancing detection accuracy, reducing false positives, and adapting to evolving malware tactics, ultimately contributing to a more secure digital environment. The main keys of the study on detecting malware threats for cybersecurity are as follows:

- Evaluates an effectiveness of various ML models for detecting malware threats.
- Provides a detailed analysis of performance metrics, enhancing understanding of model strengths and weaknesses.
- Utilizes the EMBER v2017 dataset, contributing valuable insights for future cybersecurity research.
- Demonstrates Neural Network model's supremacy in malware detection over other techniques.
- Supports the development of improved cybersecurity strategies through the implementation of advanced ML techniques.

A. Structure of the paper

Here is the breakdown of the study: Methods for identifying malicious apps on Android are covered in Section II. Section III discusses the technique. Section IV provides a summary and analysis of the experiment outcomes. In Section V, the findings and recommendations for more research are provided.

2. Literature Review

This section reviews various efforts in the literature focused on Detecting Malware Threats for Cybersecurity. It highlights key studies that explore different approaches, including dynamic, hybrid and static malware analysis techniques. A summary of the most relevant research papers on this topic is provided in Table 1.

Li et al., (2019) presents a framework for machine learning that can detect and identify DGA domains in order to mitigate the risk. In addition, develop a DNN model to effectively manage an extensive dataset that we have gradually accumulated, thereby enhancing the proposed machine learning framework. It is evident from all of our tests that the DNN model and the proposed framework are accurate. Exactingly, our results show that the framework's classification accuracy is 95.69%, the DNN model's accuracy is 90.79%, the second-level clustering accuracy is 92.45%, and the HMM prediction accuracy is 95.21%[30].

Walker et al., (2022) extrapolates the method's accuracy in detecting and classifying malware families by applying further analysis on a subset of API call sequences. Findings show that compared to earlier approaches, ELM and OS-ELM learn faster, achieving 91% accuracy in only 3 seconds because strategy outperforms others according to accuracy and training time [31].

Chaudhary et al., (2020) a study has examined a range of security risks and protective approaches, as well as open issues in the cybersecurity area for systems that identify intrusions, malware, and network anomalies using different DL and ML algorithms. Maximum accuracy of 99.90% was attained using a RBF–SVM model for intrusion detection, and 97.79% for virus identification. A DNN model achieved an accuracy of 96% when used to detect pirated software. The Seq2Seq (Sequence-to-Sequence) model had the highest accuracy at 99.90% anomaly detection in networks. Instead, if the model is based on the Deep Belief Networks (DBN), it yields an accuracy of 69.77% when it is used for anomaly detection[32].

Ghalaty and Ben Salem, (2018) offer flexible scope hierarchy that helps to sort the malware faster with its corresponding classification reflecting the priorities of the organisation. Furthermore, the paper describes our first deep neural network with the discriminating features of cyber-espionage-specific malware from the rest of the malware sample. This model was tested and validated on a balanced dataset that consists of both types of files. It had a very low false negative rate of 2.8% but in percentage of correct detections it was 97%[33].

Singh et al., (2022) offer a more defined structure to express malware quickly through a priority system of an organisation. Furthermore, we introduce the very first deep neuronal network for detecting cyber espionage-specific and overall generic malware. This model has been tested as well as validated with balanced data set of both types of files. It achieved a false negative rate of only 2.8% thus a detection rate of 97%[34].

Table 1: Comparative Study of Machine Learning Models for Cybersecurity Threat Detection

Ref	Methodology	Dataset	Result	Limitation
[30]	- Two-level model: Classifies DGA domains and clusters DGA algorithms	Real-time traffic data (1 year)	- Classification accuracy: 95.69% - DNN accuracy: 90.79%	- Real-time implementation challenges not discussed - Generalization issues with other datasets - Future work: Improve real-time detection and scalability of clustering
[31]	- Uses subset of API call sequences - ELM and OS-ELM for fast learning	API call sequences dataset	91% accuracy with just 3 seconds of learning time	- Limited evaluation on more complex datasets - Future work: Apply to more diverse datasets and improve training speed
[32]	- Uses various ML and DL algorithms - RBF-SVM for intrusion detection	Multiple security datasets	- RBF-SVM for intrusion detection: 99.90% - DNN for malware detection: 97.79	- Low DBN accuracy (69.77%) for anomaly detection - Future work: Explore alternative architectures for anomaly detection and investigate emerging cybersecurity threats
[33]	-Hierarchical framework for malware detection	Balanced dataset of malware samples	- Detection rate: 97% - False negative rate: 2.8%	- False negative rate could be a security risk - Future work: Focus on reducing false negatives
[34]	- Several supervised ML algorithms: RF, DT, ET, K-NN for IDS	CIC-IDS 2017 dataset	- Accuracy: 99% - Recall: 100% for the four classifiers	- Generalization to different network environments is uncertain - Future work: Extend to real-time environments and larger, diverse datasets

3. Research Methodology

The objective of this research is to assess and examine several ML techniques for detecting malware threats in information security systems. Therefore, in this work, the efficacy of the identified methods such as Neural Network, SVM, and RF with reference to the EMBER v2017 database is to establish the most relevant and efficient techniques of distinguishing between benign and malicious execut Table files.

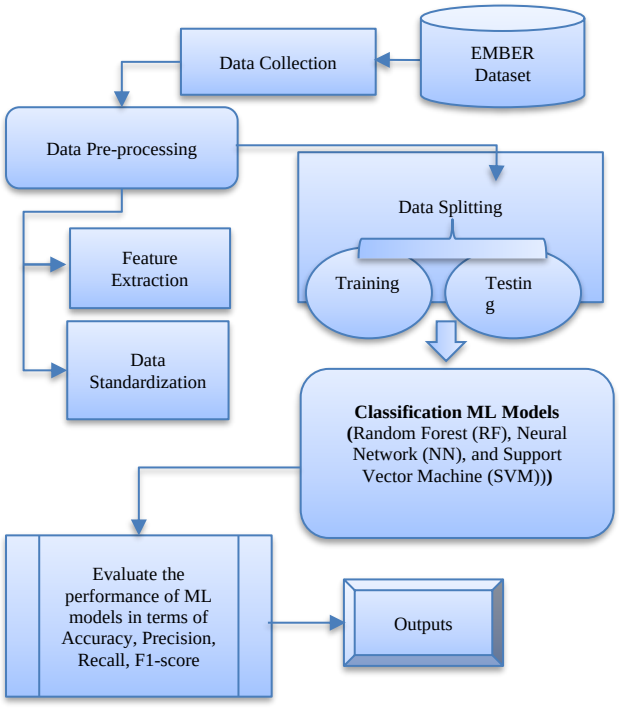

Figure 1: Methodology flowchart for Detecting Malware Threats for Cybersecurity

This evaluation will on turn aid in improving the recognition of malware thus strengthening the securities against such new emerging threats. The strategy of this study is comprised of a few key elements. An initial data collection is conducted using the EMBER v2017 dataset by comprising feature extractions from 1.1 million Windows execu Table files. Some data is then cleaned to eradicate issues such as noise and variation and some data is also made more structured. Data standardisation ensures that all the features are of the same size while feature extraction ensures that only relevant features are used for model making. After that, the dataset is divided into 20% for testing and 80% for training. Lastly, to improve malware detection capabilities, classification models like SVM, RF, and Neural Networks are used to evaluate the data and make use of their own advantages in performing classification jobs. Figure 1 depicts the flowchart of the method for Detecting Malware Threats for Cybersecurity. Every stage of the system is thoroughly explained.

The steps in the flowchart diagram are listed below.

A. Data Collection

The methodical process of obtaining and evaluating data from multiple sources to create an extensive dataset is known as data collection[35]. The EMBER v2017 dataset includes feature extractions from 1.1 million Windows execu Table files, both benign and malicious. Collected using automated analysis tools, it captures relevant metadata and behaviour patterns. This diverse and balanced dataset supports malware detection research and aids in developing machine learning models for cybersecurity.

A. Data preprocessing

The process of transforming raw data into a format that may be understood is known as data preparation[17]. Real-world data may sometimes be noisy, repetitious, erratic, and lacking. Several procedures are used in data preprocessing to assist transform unprocessed data into processed and logical form. These are the essential pre-processing methods:

1) Feature Extraction

The process of feature extraction entails locating and choosing from raw data the essential qualities or features that are most relevant to the model. In cybersecurity, features may include file size, header information, or specific patterns in execu Table binaries. In this process, complex data is transformed into a structured format that can be understood by ML algorithms.

2) Data Standardization

Data standardisation entails reducing features to a zero-mean and one-standard-deviation distribution. As a result, machine learning algorithms' performance and convergence are both enhanced, and features with smaller sizes are less likely to dominate the model.

B. Data Splitting

The dataset is split into 80:20, where 80% is employed for training the model, and 20% is reserved for testing to evaluate performance on unseen data.

C. Classification Models

Provide an explanation of the RF, NN, and SVM machine learning models in this area in order to assess how well those models identify malware threats.

1) RF

When it comes to classification and regression, RF uses the ensemble learning approach, making it a supervised learning algorithm. It runs many regression trees and then combines them into a single model to obtain greater prediction accuracy than a single tree would. During training[10], RF builds a large number of DT, and then uses the pooled forecasts of all the trees to arrive at the final prediction. By utilising RF, or random sampling with replacement, data scientists are able to lower the variance associated with algorithms, most commonly decision trees, that have a high variation. RF is a machine-learning name for bagging.

2) Support Vector Machine

One well-known ML method for classifying and regressing issues is the SVM. A number of applications have made use of SVM, such as bioinformatics and cheminformatics. Applying training data, the SVM classifier constructs a classification model. The categorisation of a randomly selected sample follows. Separating various groups using hyperplanes is the primary notion of support vector machines (SVM). For datasets that lend themselves to linear separation, SVM has proven to be remarkably accurate. On the other hand, SVM output does not support non-linear separation of separable data. After the data is transferred to a new high-dimensional space, it can be separated linearly using kernel functions. Among the most significant challenges with support vector machines (SVM) are the proper parameters and kernel function selection. [36]. SVMs are able to classify a collection of data that was originally one-dimensional in a "two-dimensional" way thanks to a mathematical concept called the kernel function.

3) Neural Network

Artificial neurones that can process numerous inputs and output a single result are used to create sophisticated structures called neural networks. A neural network's main function is to convert input into a meaningful output. A neural network typically has one or more hidden layers within of its input and output layers. It is sometimes referred to as an ANN. Figure 2 depicts the neural network's architecture.

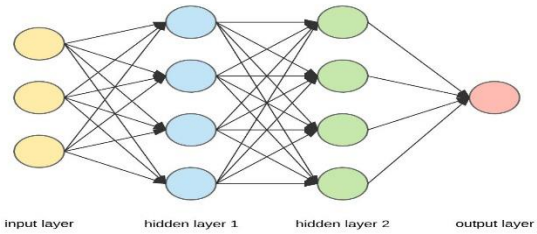

Figure 2: Architecture of Neural Network

Importantly, ANN architecture in Neural Networks works similarly to the human brain. Each neurone in a Neural Network has an effect on every other neurone because of the interconnected nature of the network. The network is able to recognise and analyse all features of the dataset, including any potential relationships between the various data elements. This is the secret behind Neural Networks' ability to uncover intricate patterns in massive datasets.

4) Trained Neural network

In our research, we developed a deep learning model for detecting malware threats in cybersecurity using the Keras library, focusing on simplicity and efficiency. Since the data was preprocessed and finely organised, our model aimed to minimise resource and time requirements by acting purely as a classifier without adding unnecessary complexity. We created a straightforward neural network consisting of dense and dropout layers. Dropout layers were used to improve regularisation and prevent overfitting by randomly deactivating neurons, while dense layers were chosen for their ability to utilise all features effectively. Two dropout layers with dropout rates set to 0.2 and 0.5 and two fully connected layers with 1500 and 1 nodes were used in the model architecture, using the Adam optimiser, 0.001 learning rate, binary cross entropy as a loss function, and trained using data in batches of 256 for 10 epochs. Therefore, using this architecture and utilising efficient methods of regularisation we were able to obtain high prediction accuracy and yet our method did not require high computation complexity that is inherent in most complicated models.

4. Result Analysis and Discussion

The machine learning experiments of this section highlighted the performance of different models for identifying malware threats in a Python simulation tool running on HP hardware and 32GB RAM. This section

of the study elaborated on the dataset characteristics, performance metrics, machine learning outcomes and a comparison as well as a discussion.

A. Data Description

In the cybersecurity sector, EMBER dataset owned by Endgame Inc. is a package that is widely used for analysis of threats, detection of risks, and their utilisation for overall improvement. Because it has turned out that it contains features that are extracted from 1.1 million Windows execu Table binary files, it is an inestimable source of information for analysis and research into malware. It contains both malicious and benign samples and there are numerous features extracted from it in order to classify and better understand the malware samples. Specifically, the EMBER v2017 dataset is employed to design and train machine learning models to improve overall threat detection through detailed information about Execu Table files, making the dataset a vital tool in enhancing cybersecurity studies. Dataset samples of EMBER 2017 are given below:

Table 2: EMBER 2017 Dataset samples distribution

Training Samples			**Testing Samples**	
Malicious	***Benign***	***Unlabeled***	***Malicious***	***Benign***
300k	300k	300k	100k	100k
Total training samples 900k			Total testing samples 200k	

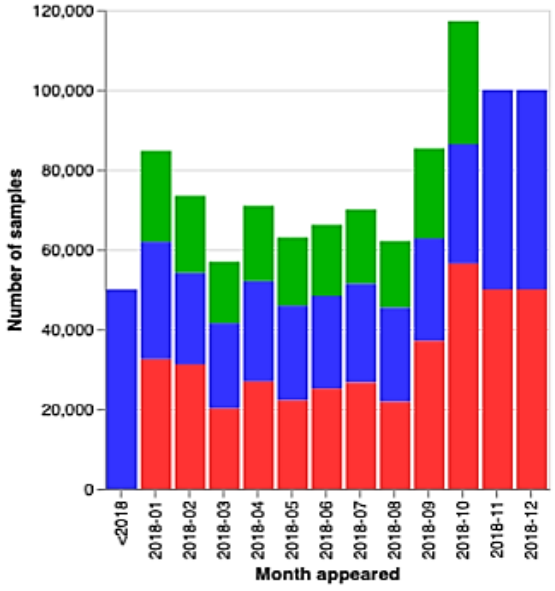

Figure 3: Distribution of PE samples in the EMBER2017 dataset.

The example in Figure 3 also illustrates that the EMBER2018 dataset is reasonably well divided between benign and malicious execu Table samples and that there are samples from each month of 2018. This distribution is a good combination that benefits the cybersecurity ML model to have benign samples, malware and unknown samples, which are present in green colour.

A. Performance Measures

Precision, recall, accuracy, and F1-score were some of the measures used to evaluate the model's performance; this allowed for a separate evaluation of each class. To provide the formulas for their evaluation, these parameters are given below:

1) Confusion Matrix

The confusion matrix's visual representations are shown in Figure 4. The confusion matrix provides insight into the model's performance; it is a matrix with dimensions N x N, where N is the total number of target classes. The diagonal numbers reflect the amount of correct classifications; glancing at them makes it easy to evaluate the model's accuracy and identify possible misclassification zones.

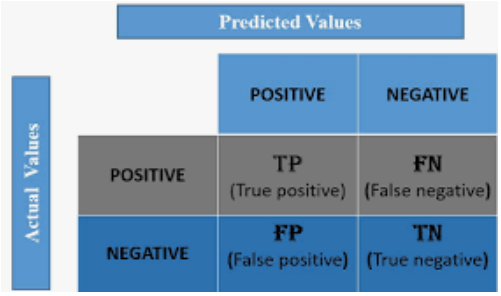

Figure 4: Visual representation of confusion matrix.

A terms used in a confusion matrix are as follows:

- **TP: True Positive:** An actual number was positive, while a model expected a positive value.
- **FP: False Positive**: The forecast is both incorrect and optimistic. Alternatively known as the Type 1 mistake.
- **FN: False Negative:** Both the result and the forecast are incorrect. (Also known as the Type 2 mistake)
- **TN: True Negative:** A negative result was obtained, which was contrary to what the model had predicted.

2) Accuracy

Accuracy is how many test dataset predictions your model made correctly. Basic model performance metrics include accuracy. Lack of balance in datasets makes accuracy a poor Metrix. It is given as in the Equation (1):

$$Acuracy = \frac{TP+TN}{TP+TN+FN+FP} \quad (1)$$

3) Precision

Precision shows how many accurately expected cases were positive. As shown in the Equation(2):

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

4) Recall

Recall shows how many positive cases our model predicted accurately. This metric is useful when FN

overcomes FN. It is mathematically given by Equation (3):

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

5) F1-score

Precision and recall are harmonically averaged to give the F1-Score, which offers a thorough comprehension of these two metrics. The highest value is achieved when the recall and precision are equivalent. Mathematically, it is given as in the Equation (4):

$$F1 - score = 2 * \frac{precision*recall}{precision+recall} \quad (4)$$

The following section discusses the experimental results of Neural Network models for malware threat detection.

B. Experiment Results

This section provides the experimental results derived from using deep and machine learning models based on Detecting Malware Threats for Cybersecurity. The results of Table are illustrated through figures and Table 3, offering an in-depth perspective on the performance and strengths of each model.

Table 3: Results of Neural Network model

Performance Measures	Neural Network Model
Accuracy	97.53
Precision	98.85
Recall	95.40
F1-score	95.23

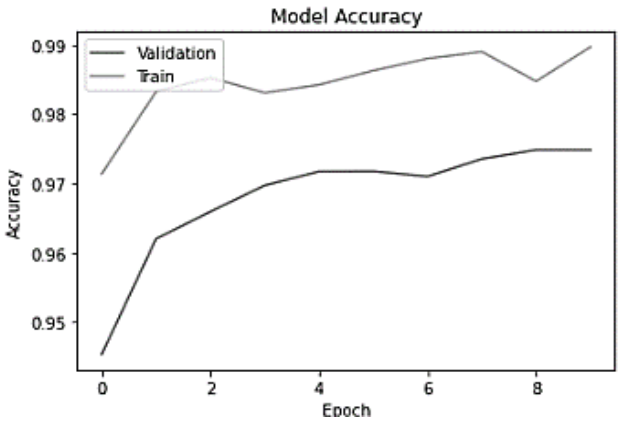

Figure 5: Accuracy graph for Neural Network model.

Figure 5 displays the model's accuracy for the training and validation datasets over a span of 10 epochs. The model constantly learns from the training data with a training accuracy close to 0.99, demonstrating its effectiveness. Validation accuracy is slightly lower, ranging from 0.97 to 0.98, and it stabilises after the initial few epochs, indicating that there is minimal overfitting and excellent generalisation.

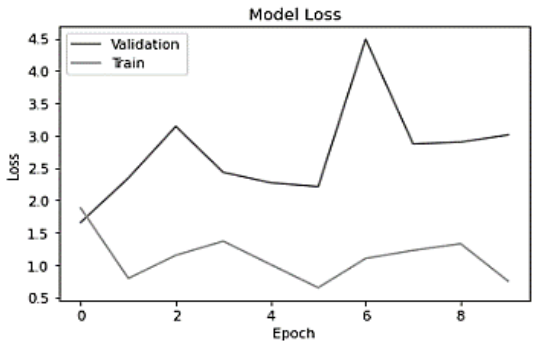

Figure 6: Loss graph for Neural Network model.

Figure 6 shows the model loss across 10 epochs for both the training and validation datasets. The model's learning progress is demonstrated by the steady decrease in the training loss, while the validation loss is more volatile, with a no Table rise at epoch 6. This variance in validation loss might indicate some degree of instability or overfitting at certain points in training.

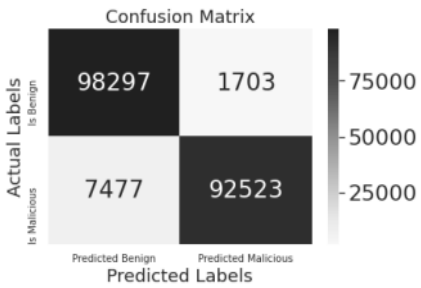

Figure 7: Confusion matrix of Neural Network model

Figure 7 representing the confusion matrix of model's classification performance between benign and malicious labels. Out of a total dataset, the model correctly predicts 98,297 benign samples and 92,523 malicious samples, while it misclassifies 1,703 benign samples as malicious and 7,477 malicious samples as benign. This shows that the model successfully differentiates between the two groups with few erroneous predictions.

C. Comparative analysis

The following Table 4 summarises the results of numerous DL and ML models that were used to detect malware using the EMBER dataset for cybersecurity. To evaluate the efficacy of enhanced ML models according to F1-score, Accuracy, Precision, and Recall measures. These assessment factors are outlined below.

Table 4: Comparison between various model for Detecting Malware Threats for Cybersecurity

Models	Accuracy	Precision	Recall	F1-Score
RF [37]	84.3	85.3	84.3	83.6
SVM [38]	91	91	95	95
Neural Network	97.53	98.85	95.40	95.23

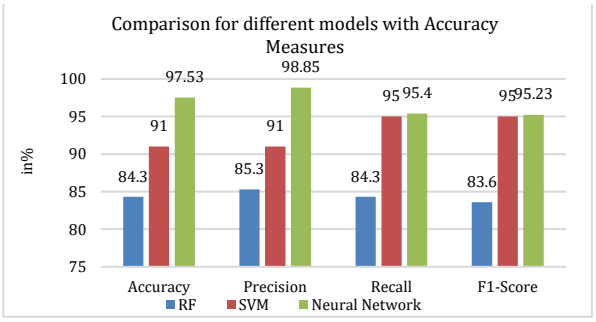

Figure 8: Comparison for different models for Accuracy.

Figure 8 presents a comparison of different models for detecting malware threats in cybersecurity based on various performance metrics: accuracy, precision, recall, and F1-score. In the present work, the Random Forest (RF) model yielded 84.3% accuracy and 85.3% precision, 84.3% recall, and 83.6% F1 score. This is a mean performance which, however, is generally considered satisfactory. On the other hand, the proposed Support Vector Machine (SVM) model yielded 91% accuracy, 91% precision, 95% of recalls, and 95% of F1 score, which paves way to a significantly improved model in MALWARE threatening detection since it was shown to demonstrate higher recall data. Neural Network model showed the best performance with accuracy 97.53% and Precision 98.85%, Recall 95.40%, and F1 score 95.23%. This partly implies that the Neural Network is very good in eliminating false positives while also presenting very good balance of both precision and recall score, making it the best over all the compared models for malware detection.

Conclusion and Future Work

Dangerous program files are recognised as malicious software that intends to harm different types of devices, networks, and servers. Malware is spreading at an alarming rate on a daily basis due to the proliferation of both gadgets and technology. Malicious assaults, the majority of which target organisations, customers, enterprises, etc., are increasing at a rate directly equal to the proliferation of gadgets and computers and technological advancements. Utilising the EMBER v2017 dataset, Several ML models were proficiently evaluated in this study for their ability to identify malware hazards within cybersecurity frameworks. According to recall, precision, accuracy, and F1-score, the Neural Network model outperformed the other competing models (RF and SVM) with 97.53% accuracy. Machine learning techniques have the potential to strengthen cybersecurity defences against evolving threats and enhance malware detection capabilities, as these results underscore.
Research in the future might look at more advanced ML approaches, like ensemble methods and DL architectures, to see whether they might boost detection performance. Additionally, incorporating real-time data analysis and adapting models to recognise emerging malware patterns could significantly bolster defences. Investigating the integration of threat intelligence and behavioural analysis may also provide deeper insights into malware detection, allowing for more proactive cybersecurity strategies.

references

[1] T. McIntosh, J. Jang-Jaccard, P. Watters, and T. Susnjak, "The Inadequacy of Entropy-Based Ransomware Detection," in *Communications in Computer and Information Science*, 2019. doi: 10.1007/978-3-030-36802-9_20.
[2] D. Geer, E. Jardine, and E. Leverett, "On market concentration and cybersecurity risk," *J. Cyber Policy*, 2020, doi: 10.1080/23738871.2020.1728355.
[3] Mani Gopalsamy, "An Optimal Artificial Intelligence (AI) technique for cybersecurity threat detection in IoT Networks," *Int. J. Sci. Res. Arch.*, vol. 7, no. 2, pp. 661–671, Dec. 2022, doi: 10.30574/ijsra.2022.7.2.0235.
[4] P. Khare, "The Impact ofAI on Product Management:A Systematic Review and Future Trends," *Int. J. Res. Anal. Rev.*, vol. 9, no. 4, pp. 736–741, 2022.
[5] S. Pandey, "Transforming Performance Management Through Ai: Advanced Feedback Mechanisms, Predictive Analytics, And Bias Mitigation In The Age Of Workforce Optimization," *Int. J. Bus. Quant. Econ. Appl. Manag. reseacrh*, vol. 6, no. 7, pp. 1–10, 2020.
[6] R. Arora, S. Gera, and M. Saxena, "Mitigating security risks on privacy of sensitive data used in cloud-based ERP applications," *Proc. 2021 8th Int. Conf. Comput. Sustain. Glob. Dev. INDIACom 2021*, no. March 2021, pp. 458–463, 2021, doi: 10.1109/INDIACom51348.2021.00081.
[7] M. S. Rajeev Arora, "Applications of Cloud Based ERP Application and how to address Security and Data Privacy Issues in Cloud application," *Himal. Univ.*, 2022.
[8] S. Papastergiou, H. Mouratidis, and E. M. Kalogeraki, "Cyber security incident handling, warning and response system for the european critical information infrastructures (cyberSANE)," in *Communications in Computer and Information Science*, 2019. doi: 10.1007/978-3-030-20257-6_41.
[9] V. K. Y. Nicholas Richardson, Rajani Pydipalli, Sai Sirisha Maddula, Sunil Kumar Reddy Anumandla, "Role-Based Access Control in SAS Programming: Enhancing Security and Authorization," *Int. J. Reciprocal Symmetry Theor. Phys.*, vol. 6, no. 1, pp. 31–42, 2019.
[10] J. Thomas, "Enhancing Supply Chain Resilience Through Cloud-Based SCM and Advanced Machine Learning: A Case Study of Logistics," *J. Emerg. Technol. Innov. Res.*, vol. 8, no. 9, 2021.
[11] S. G. Priya Pathak, Akansha Shrivastava, "A survey on various security issues in delay tolerant networks," *J Adv Shell Program.*, vol. 2, no. 2, pp. 12–18, 2015.
[12] S. Dixit, P. Pathak, and S. Gupta, "A novel approch for gray hole and black hole detection and prevention," in *2016 Symposium on Colossal Data Analysis and Networking, CDAN 2016*, 2016. doi: 10.1109/CDAN.2016.7570861.
[13] S. Pandey, "Leveraging Workday For Effective Covid-19 Vaccination Tracking: Integrating Custom Objects And Security Features In Human Capital Management Systems," *Int. J. Bus. Quant. Econ. Appl. Manag. reseacrh*, vol. 7, no. 1, pp. 56–63, 2021.
[14] Mani Gopalsamy, "A review on blockchain impact on in cybersecurity: Current applications, challenges and future trends," *Int. J. Sci. Res. Arch.*, vol. 6, no. 2, pp. 325–335, Aug. 2022, doi: 10.30574/ijsra.2022.6.2.0146.

[15] M. E. O'connell, "Cyber security without Cyber war," *J. Confl. Secur. Law*, 2012, doi: 10.1093/jcsl/krs017.
[16] R. Bishukarma, "Adaptive AI-Based Anomaly Detection Framework for SaaS Platform Security," *Int. J. Curr. Eng. Technol.*, vol. 12, no. 07, pp. 541–548, 2022, doi: https://doi.org/10.14741/ijcet/v.12.6.8.
[17] K. Patel, "Quality Assurance In The Age Of Data Analytics: Innovations And Challenges," *Int. J. Creat. Res. Thoughts*, vol. 9, no. 12, pp. f573–f578, 2021.
[18] M. R. S. and P. K. Vishwakarma, "An Efficient Machine Learning Based Solutions for Renewable Energy System," *Int. J. Res. Anal. Rev.*, vol. 9, no. 4, pp. 951–958, 2022, [Online]. Available: https://www.ijrar.org/papers/IJRAR22D3208.pdf
[19] V. V. Kumar, A. Sahoo, and F. W. Liou, "Cyber-enabled product lifecycle management: A multi-agent framework," in *Procedia Manufacturing*, 2019. doi: 10.1016/j.promfg.2020.01.247.
[20] M. Gopalsamy, "Scalable Anomaly Detection Frameworks for Network Traffic Analysis in cybersecurity using Machine Learning Approaches," *Int. J. Curr. Eng. Technol.*, vol. 12, no. 06, pp. 549–558, 2022.
[21] R. P. Vamsi Krishna Yarlagadda, "Secure Programming with SAS: Mitigating Risks and Protecting Data Integrity," *Eng. Int.*, vol. 6, no. 2, pp. 211–222, 2018, doi: 10.18034/ei.v7i2.711.
[22] S. Bauskar, "Business Analytics in Enterprise System Based on Application of Artificial Intelligence," *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 04, no. 01, pp. 2582–5208, 2022, doi: 10.56726/IRJMETS18127.
[23] M. Gopalsamy, "Advanced Cybersecurity in Cloud Via Employing AI Techniques for Effective Intrusion Detection," *Int. J. Res. Anal. Rev.*, vol. 8, no. 1, pp. 187–193, 2021, [Online]. Available: https://www.ijrar.org/papers/IJRAR21A1737.pdf
[24] M. Gopalsamy, "Artificial Intelligence (AI) Based Internet-ofThings (IoT)-Botnet Attacks Identification Techniques to Enhance Cyber security," *Int. J. Res. Anal. Rev.*, vol. 7, no. 4, pp. 414–420, 2020, [Online]. Available: https://www.ijrar.org/papers/IJRAR2AA1742.pdf
[25] P. Khare, "Signature-Based Biometric Authentication: A Deep Dive Into Deep Learning Approaches," *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 04, no. 08, pp. 2414–2424, 2022, [Online]. Available: https://www.irjmets.com/uploadedfiles/paper//issue_8_august_2022/29522/final/fin_irjmets1723873974.pdf
[26] T. Hey, S. Tansley, and K. Tolle, "The Fourth Paradigm," *Data-Intensive Sci. Discov. Microsoft Res.*, vol. 99, no. 8, p. 287, 2009.
[27] Mani Gopalsamy, "Enhanced Cybersecurity for Network Intrusion Detection System Based Artificial Intelligence (AI) Techniques," *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 12, no. 01, pp. 671–681, Dec. 2021, doi: 10.48175/IJARSCT-2269M.
[28] S. G. Jubin Thomas, Piyush Patidar, Kirti Vinod Vedi, "Predictive Big Data Analytics For Supply Chain Through Demand Forecastin," *Int. J. Creat. Res. Thoughts*, vol. 10, no. 06, pp. h868–h873, 2022.
[29] S. Anwar *et al.*, "From intrusion detection to an intrusion response system: Fundamentals, requirements, and future directions," *Algorithms*. 2017. doi: 10.3390/a10020039.
[30] Y. Li, K. Xiong, T. Chin, and C. Hu, "A Machine Learning Framework for Domain Generation Algorithm-Based Malware Detection," *IEEE Access*, 2019, doi: 10.1109/ACCESS.2019.2891588.
[31] A. Walker, R. M. Shukla, T. Das, and S. Sengupta, "Ohana Means Family: Malware Family Classification using Extreme Learning Machines," in *Proceedings - IEEE Consumer Communications and Networking Conference, CCNC*, 2022. doi: 10.1109/CCNC49033.2022.9700583.
[32] H. Chaudhary, A. Detroja, P. Prajapati, and P. Shah, "A review of various challenges in cybersecurity using artificial intelligence," in *Proceedings of the 3rd International Conference on Intelligent Sustainable Systems, ICISS 2020*, 2020. doi: 10.1109/ICISS49785.2020.9316003.
[33] N. F. Ghalaty and M. Ben Salem, "A Hierarchical Framework to Detect Targeted Attacks using Deep Neural Network," in *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*, 2018. doi: 10.1109/BigData.2018.8622131.
[34] A. P. Singh, S. Kumar, A. Kumar, and M. Usama, "Machine Learning based Intrusion Detection System for Minority Attacks Classification," in *Proceedings of International Conference on Computational Intelligence and Sustainable Engineering Solution, CISES 2022*, 2022. doi: 10.1109/CISES54857.2022.9844381.
[35] A. P. A. Singh, "Strategic Approaches To Materials Data Collection And Inventory Management," *Int. J. Bus. Quant. Econ. Appl. Manag. Res.*, vol. 7, no. 5, 2022.
[36] A. Tharwat, A. E. Hassanien, and B. E. Elnaghi, "A BA-based algorithm for parameter optimization of Support Vector Machine," *Pattern Recognit. Lett.*, vol. 93, pp. 13–22, 2017, doi: 10.1016/j.patrec.2016.10.007.
[37] R. Vinayakumar, M. Alazab, S. Kp, P. Poornachandran, and S. Venkatraman, "Robust Intelligent Malware Detection Using Deep Learning," *IEEE Access*, vol. PP, no. March 2020, p. 1, 2019, doi: 10.1109/ACCESS.2019.2906934.
[38] L. Ghouti and M. Imam, "Malware classification using compact image features and multiclass support vector machines," *IET Inf. Secur.*, vol. 14, no. 4, pp. 419–429, 2020, doi: 10.1049/iet-ifs.2019.0189.