Advanced Sciences and Technology Journal

ASTJ vol. 3 no. 1 (2026) P1035
10.21608/ASTJ.2025.346317.1035
https://astj.journals.ekb.eg/

Transforming Network Security: The Role of Integrated AI in Intrusion Detection Systems

Soha Safwat[1,*], Renad Samy[2], Mohamed Tarek[2], Marina Magdy[2], Mohamed Atef[2]

[1]*Computer Science, Faculty of Computer and Information Systems, Egyptian Chinese University, Cairo, Egypt*
[2]*Software Engineering, Faculty of Engineering and Technology, Egyptian Chinese University, Cairo, Egypt*
** ssafwat@ecu.edu.eg*

ARTICLEINFO

Article history:
Received 21 December 2024
Revised 25 April 2025
Accepted 22 August 2025
Available online 14 September 2025

Handling Editor: Prof. Dr. Mohamed Talaat Moustafa

Keywords:
Malware,
CNN,
RNN,
Machine Learning,
Deep Learning.

ABSTRACT

The concept of Intrusion Detection Systems becomes very important in light of highly sophisticated and pervasive cyber threats. AI had finally become an agent of change by improving IDS systems through the application of sophisticated models of machine learning and deep learning in real-time complex security threat detection, analysis, and response. Besides, hybrid architectures that involve the integration of CNN and LSTM further apply AI-driven systems to make intrusion detection effective due to the capability to analyse even complex network traffic patterns. Combining spatial feature extraction with temporal sequence analysis makes it possible to precisely identify both known and unknown threats with high precision. Complex modern network environments detected the presence of obfuscated attacks and zero-day, which conventional methods could not detect. The incorporation of AI in IDS significantly enhances the detection accuracy while reducing false positives and offering adaptability in dynamic network conditions. These make the systems practical to run in real time, since analysis can be done without complex pre-processing of raw data; hence, they are indispensable for modern cybersecurity strategies. However, many challenges remain yet to be addressed regarding network traffic variability management, class imbalance handling, and false alarms minimization in an operational environment. Only continuous learning with model updates in this dynamic cyber threat environment will keep performance sustained. The present research has demonstrated the transformative role of AI in intrusion detection by underpinning its potential to revolutionize network security through proactive, adaptive, and resilient defences against an ever-evolving threat landscape.

1. Introduction

In the present digital ecosystem, where everything is connected, device vulnerability to sophisticated cyberattacks poses severe risks both at the organization and individual levels. The threats of ransomware, phishing, and DDoS attacks are evolving day by day by exploiting vulnerabilities in the security of devices and networks. For such challenges, effective monitoring systems must be put in place. Monitoring of devices has a very crucial role in the detection of anomalies and unusual activities, providing early warnings regarding imminent dangers. Monitoring systems identify suspicious patterns by constantly observing and analysing device behaviour, flag potential breaches, and initiate proper countermeasures in order to limit the risks.

Traditional IDS approaches have been based, for a long time, on signature-based or rule-based methods of detection for malicious activities. While effective within the realm of an already known threat, intrusion detection systems intrinsically cannot hold up to novel, obfuscated, or zero-day attacks. Furthermore, the ever-increasing scale and complexity of network traffic overwhelms conventional systems, with even higher false alarm rates and reduced reliability in dynamic environments. It is in view of this that Artificial Intelligence has presented a changing solution to deal with these problems by promising an enhanced intrusion detection capability: intelligent, adaptive, automated systems.

AI-powered intrusion detection systems combine ML and DL to form perfect analyzers of network traffic patterns for the detection of anomalies with unparalleled accuracy. These systems use various advanced algorithms, which track the real-time data streams, learn from historical and live traffic, and sort out what is normal and what is malicious. Traditional techniques are at odds with intrusion detection system designs using AI, as these need no extensive pre-defined Signature eliciting or manual updating of rules; instead, they are inherently more agile in identifying unknown threats.

Among the most promising developments in this domain are those integrating hybrid models such as CNN and LSTM. Although CNNs are relatively good at data features extraction with a spatial dimension, the major focus of LSTMs is on temporal dependencies, making them the most suitable for complex and sequential pattern analyses of network traffic. Cumulatively, these architectures allow AI-powered IDS to identify various attack vectors that remain out of the reach of conventional systems, such as advanced persistent threats and polymorphic malware.

Beyond the powers of mere detection, AI tends to enhance the capabilities of monitoring through continuous network behavioral oversight. AI-powered systems can analyze enormous volumes of data in real time and pick out anomalies indicative of potential intrusions. It ensures timely detection and response to minimize attacker dwell time within the networks through proactive attacks. The system lightens the load on security teams through automation of routine tasks, prioritization of alerts, and actionable insights with deep analytics.

Certain challenges appear in the deployment of AI-driven IDS: First, the dynamic nature of network traffic is bound to keep the models retraining for high-accuracy detection. First, issues such as class imbalance in data-the normal instances outnumber the malicious instances manyfold-can skew the detection rate. Second, false-positive generation remains a persistent problem because too many alerts could overwhelm security operations teams and dilute their effectiveness. Scalability is another critical factor, as modern networks produce ever-growing amounts of data, necessitating high-performance computing resources to support AI-driven solutions.

2. Techniques and Strategies

The advancements in both machine learning and deep learning techniques have brought about an immense change in the detection of malicious attacks in networking. Unsupervised Learning, Deep Learning, and Natural Language Processing (NLP) are primary methods. The approaches are powerful tools for comprehending complex datasets and for eliciting hidden patterns in user behavior[1].
Examples of these unsupervised learning methods are Local Outlier Factor (LOF), Isolation Forest (IF), One-Class Support Vector Machine (OCSVM), and Principal Component Analysis (PCA). These methods assist in faultlessly detecting anomalies without needing labelled data. For detecting anomalies likely connected with a breach, the deep learning techniques-motivated mostly by auto-encoders-are effective at reconstructing patterns of what is regularly expected. NLP is a burgeoning area for textual data analytics- especially around insider threat detection-combining language understanding and anomaly detection, thus effectively addressing the increasing sophistication of threats.[2]

Besides basic approaches mentioned, adroit techniques like Seq2Seq models have shown their prowess in capturing

network anomalies while adversarial training has demonstrated its capability of improving the robustness of ML models under adversarial attack. The defensive mechanisms for protecting the models were then summarized into several classes: these classes give an account of security mechanisms to assess the vulnerability of a model, countermeasures in the training phase to secure the processes and data of training, and countermeasures during testing to protect a model at inference time. Engagement with the security and privacy of the data under encryption, as well as integrity maintenance for the data during training, are highly prioritized issues in recent years.[3][4]

These techniques show the developing role of AI-afforded solutions in cybersecurity challenges, with strong means to detect, defend, and mitigate security threats in an increasingly complex digital world.

2.1. Machine Learning (ML)

Machine Learning is critical in most integrated AI cybersecurity systems, including IDS. Through the analysis of network traffic and system behaviour, ML algorithms identify patterns and anomalies that indicate a potential security threat. Supervised learning models learn from labelled datasets to recognize known attack types, whereas unsupervised models uncover novel or unknown threats by recognizing deviations from normal behaviour. With continuous learning, ML equips IDS with the ability to adapt and develop over time, providing real-time proactive defence against evolving cyber-attacks that showed in figure 1 [5].

2.1.1. Supervised learning

2.1.1.1. Learning rate (LR)

Logistic Regression-LR is a statistical model used in binary classification cases where the goal is the prediction of the probability belonging to one of two classes. It outputs probabilities for estimation, taking as input the logistic function of a linear combination of features. Despite being so simple, LR remains effective in many tasks-such as spam detection and fraud detection. While it does quite well on linearly separable data, it can be far less effective on more complex, nonlinear problems without feature engineering or other modifications[6].

2.1.1.2. Random Forest (RF)

RF (Random Forest) is a model assembling methodology in machine learning wherein several decision trees are combined with the goal of improving the classification accuracy and robustness of any intrusion detection system. It creates a set of decision trees based on random subsets of features and gives a set of diversified models that vote on the final classification. While RF has a resistance to overfitting and works nicely when the size of datasets becomes huge, it results in consuming more memory and trains relatively slow. While it performs cyber threat identification well, perhaps this is not as flexible as in deep learning models [5][6].

2.1.1.3. Gradient Boosting Machine (GBM)

GBM (Gradient Boosting Machine) is an ensemble model construction technique that builds models sequentially, one following the other, each adjusting for errors of the previous models. It reduces bias and dispersion by iteratively minimizing a loss function and, therefore, increases generalizing performance. GBM is capable of handling extremely complex datasets with high dimensionality of features and proving very effective in fraud detection cases or recommendation systems. However, it may have problems of overfitting if not tuned properly and may require a little careful optimization to yield good performance [6].

2.1.1.4. Support Vector Machine (SVM)

The Support Vector Machine is a supervised machine learning algorithm used for classification and regression tasks. It works by finding the optimal hyperplane which can separate various classes in some

high-dimensional space. It works to maximize the margin between classes, ensuring that the model will generalize well. SVM does both linear and non-linear classification using kernel functions, which transform the data. SVM is known for its robustness, especially in high-dimensional spaces, and in those cases where the number of features exceeds the number of samples [5].

Support Vector Machines are particularly effective in intrusion detection tasks, especially the RBF kernel. SVM seeks an optimal hyperplane to separate data points from different classes so that it can generalize well with a maximum margin. The RBF kernel allows SVM to handle nonlinear relationships between features, thus being ideal for complex datasets. RBF-SVM has been used in intrusion detection and has given very impressive results with accuracy as high as 99.90%, making it one of the most reliable methods for detecting security threats [4].

2.1.1.5. K-Nearest Neighbor (KNN)

k-NN (k-Nearest Neighbours) is a non-parametric and instance-based machine learning algorithm used in classification and regression. Intrusion detection using k-NN classifies a data point with the majority class of the nearest neighbours. The method has an intuitive sense and is straightforward to implement, but usually computationally expensive, especially when there are large datasets. k-NN performance also heavily relies on the type of distance metric and the value of k. While it is a simple method, complex cybersecurity environments and evolving threats may not be performed well within[5].

2.1.1.6. Decision Tree (DT)

Decision Trees are a nonlinear, machine learning algorithm applied both for classification and regression problems. In cybersecurity, decisions trees create a model that makes a split of the data into branches based on features values, leading to an eventual classification at the leaf nodes. While easy to interpret and understand, Decision Trees are prone to overfitting, especially if the data is noisy or with high dimensions. Decision trees will be pretty useful in simple scenarios in intrusion detection systems, while for complex, ensemble methods like Random Forest or deep learning models might turn out to be more appropriate[5].

2.1.2. Unsupervised learning

2.1.2.1. Navie Bayes (NB)

Naive Bayes is a probabilistic classifier based on Bayes' Theorem with the independence of features. It classifies events in intrusion detection by computing the likelihood of an attack given the observed features. It is fast, simple, and works well for high-dimensional data. However, this often-violated assumption of feature independence in real-world cybersecurity data restricts its performance when the features are correlated. Despite this, Naive Bayes is a good choice for fast initial threat detection when the computational resources are limited[5].

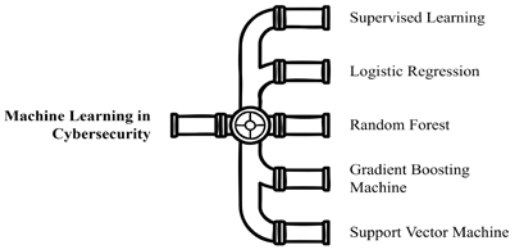

Fig.1: Exploring Machine Learning in Cybersecurity

2.2. Deep Learning (DL)

2.2.1. Long Short-Term Memory (LSTM)

LSTM-RNN is the Long Short-Term Memory (LSTM) Recurrent Neural Network (RNN), LSTMs are a type of RNN designed to learn from sequences of data making them particularly effective for tasks where context and order matter as intrusion detection in network traffic, its architecture makes the model remember information for long periods which is crucial for identifying patterns in network traffic over time. The training process has two main phases, the first phase was the forward propagation which calculates the output values based on the input data. The LSTM processes the input sequences and generates predictions regarding whether the traffic is normal or an intrusion. the second phase is back propagation, in this stage, the model will update its weights using the errors calculated from the predictions[5], [7]. Recurrent Neural Network and Long Short-Term Memory networks are designed to handle sequences of data. Such networks can, therefore, help analyse temporal patterns, which the network traffic usually evidences over time. Specifically, in IoT attack detection, it can easily capture the temporal dependency in traffic data using LSTM. Also, in addition with Bidirectional LSTMs, they increase accuracy by taking into account the events which occur both in the past and the future. But usually, they are so computationally expensive and suffer the vanishing gradient problem if sequences are long[8].

2.2.1.1. Convolutional Neural Network and Long Short-Term Memory (CNN-LSTM)

This method is a hybrid model combining CNN and LSTM. one of the models is designed to effectively analyze real-time HTTP traffic without the need for pre-feature extraction, allowing it to process all strings directly making them achieve better detection rates, but it had a problem that their system required careful event profiling and data aggregation to handle the large volume of security events [5][7].

Although convolutional neural networks are known to extract hierarchical features in a spatial manner from images, they also have their place in network traffic. Using raw traffic data transformed into a grid-like representation, CNNs can automatically learn the spatial patterns. As such, they are quite good at feature extraction and classification, particularly in multiclass problems with large datasets and a potential automatic detection of relevant features. However, CNNs may perform poorly in identifying rare attack types, such as U2R attacks, because of the underrepresentation in the training data[8].

CNN, LSTM, and the hybrid form of these have turned out to be strong solutions for malware detection. CNNs are very good at finding structural patterns in malware, particularly in image-based representations, by extracting meaningful features from spatial data. On the other hand, LSTMs are good at capturing long-term dependencies in sequential data, such as system or API call patterns during dynamic analysis. The hybrid CNN-LSTM uses the strengths of CNN for spatial analysis and those of LSTM for temporal sequence modelling, making it suitable for both behavioural analysis and image-based categorization of malware[9].

2.2.2. Fully Connected Neural Network (FCNN)

FCNN is one kind of deep learning that usually detects the cyber threats by learning complicated patterns of event data. In FCNN, each neuron is normally connected to the neurons of other layers, through which a model can carve out intimate relationships among factors. Therefore, cybersecurity may also apply FCNN with the

identification of the profile and prediction of anomalies of events. The FCNNs, trained on a vast amount of network traffic or system events, will classify malicious activities effectively, providing a strong detection mechanism [5].

2.2.3. Artificial Neural Network (ANN)

A modified version of the ANN was designed to provide a specific input layer, hidden layers, and output layer. The number of units in the input layer corresponds to the feature number in the training dataset The hidden layers utilized sufficient units along with activation functions such as ReLU (Rectified Linear Unit) to speed up learning. In the output layer, a sigmoid activation function is used, suitable for binary classification tasks-distinguishing between legitimate and malicious activities. The model is trained on a preprocessed dataset that can be effectively learned from; this includes normalization of data using a standard scaler and appropriate encoding of the target labels that helps in adjusting the range of the dataset features. This step is essential for improving the sensitivity of the ANN to subtle patterns in the input data. This training is essential to ensure that the model generalizes well on unseen data [10].

Deep Belief Networks (DBN), Hybrid Approaches, Variational Auto-Encoders (VAE), and Graph Neural Networks (GNN) / Recurrent Neural Networks (RNN): The DBN is a multi-layer network applied to the effective real-time detection of malicious traffic. It leverages the power of unsupervised learning for feature extraction, besides being fine-tuned under supervision. Hybrid approaches with combined techniques such as auto-encoder, CNNs, and LSTM can work in solving complex issues concerning high dimensionality and enhance accuracy in anomaly detection. Variational Auto-Encoders are good at learning complex patterns from imbalanced datasets and, therefore, have a wider application in outlier detection. Besides, GNN-RNN collaborations help in both spatial and temporal dependency capture in anomaly detection and further enhance the malicious pattern identification capabilities in dynamic settings. Altogether, advanced techniques like these offer robust and accurate frameworks to handle cyber threats that are also evolving continuously showed in figure 2 [11].

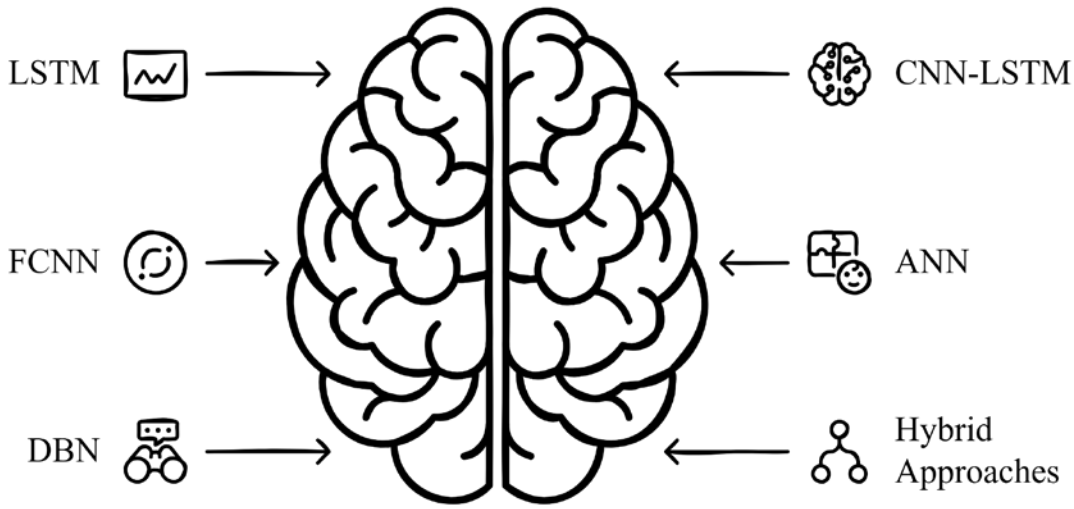

Fig.2: Deep Learning Models for Cybersecurity

2.3. Generative Adversarial Domain (GAD & GANs)

2.3.1. TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency) is a statistical method for transforming raw data, such as event profiles, into numerical vectors suitable for machine learning analysis. In cybersecurity, TF-IDF helps to quantify the importance of terms or features in a dataset relative to the whole collection of event logs. TF-IDF improves the input data that is fed into deep learning models by weighing terms on their frequency within a document and across documents, thus enhancing the capability of distinguishing between benign and malicious activities [5].

2.3.2. Seq -2- Seq

The Seq2Seq model is widely used in natural language processing and has also been used with great success in anomaly detection in cybersecurity for detecting network anomalies. A Seq2Seq model learns complex patterns in sequential data, such as network traffic or event logs, using an encoder-decoder architecture. This provides it with the capability of modelling time-series data with great efficiency for the identification of subtle anomalies in real time. Seq2Seq models have reached an accuracy rate of as high as 99.90%, making them dependable for intrusion detection and thus securing the networks[4].

2.3.3. Adversarial Training

The training methodology that makes machine learning models robust by exposing them to adversarial examples in the training phase is termed adversarial training. Such examples are crafted on purpose so as to deceive or confuse a model and represent an attack on it. When these examples are added into training, models learn how to detect and prevent such malicious input from happening. This defensive mechanism is a key factor in strengthening machine learning models in cybersecurity to resist adversarial attacks and to be generally more reliable in dynamic environments[4].

3. Datasets

3.1. KDD CUP 99 Dataset

Like the NSL-KDD dataset, this dataset also finds broad applications in intrusion detection research. It acts like a benchmark in comparing the intrusion detection performance of various machine learning models[4].

3.1.1. NSL KDD Dataset

NSL-KDD is one of the most popular and widely used datasets in network intrusion detection, and it is widely used for training and testing IDSes. Containing labelled instances of both normal and abnormal network behaviours, a wide range of various types of attacks are proposed as an appropriate benchmark in order to evaluate the performance of the IDS [12]. Basically, this is the revisited version of the benchmark dataset from the original KDDCUP99, overcoming several statistical faults of its predecessors. Hence, the dataset is applicable as a robust standard for the performance evaluation of IDSs [5][13]. The study implements four different datasets for the IDS evaluations: NSL-KDD, which is a benchmark used for evaluating intrusion detection systems [10][2]; CICIDS2017, another benchmark dataset for assessments of cybersecurity models ; and two field-collected datasets; however, specific details about the latter have not been mentioned here [10]. Among these, the NSL-KDD dataset is highly relevant to research into the area of IDS, since it is one of the most employed datasets for assessing different

IDS methods. For example, models such as the Seq2Seq model have achieved a high accuracy of 99.90% when tested on this dataset [4].

3.2. CIC-IDS 2017 & CSIC-2010 Dataset

The CICIDS2017 dataset belongs to the dataset collection from the Canadian Institute for Cybersecurity and finds wide applications in IDS-related scenarios to represent real-world scenarios for evaluating intrusion detection system performance[2][5]. Besides it, the NSL-KDD dataset is a well-known benchmark dataset for intrusion detection, and CICIDS2017 too serves as a benchmark for evaluating cybersecurity models. Also, two datasets were gathered from the field, though details about these datasets are not specified [10]. Other datasets include CSIC-2010 and CICIDS2017, public datasets used for the training and verification of IDS models. These hold real-time data of HTTP traffic, hence important in evaluating the real time performance of models [7][3]. The diversity of the testing datasets available for testing proves how much facilities are available to study intrusion detection [4][13].

3.3. Security Operation Center (SOC) Dataset

The Security Operation Centre dataset is an aggregated collection of network and security-related events from organizational environments in the real world. This dataset mostly contains logs from firewalls, intrusion detection systems (IDS), and endpoint detection systems, alongside event correlations. SOC datasets are useful to model operational cybersecurity scenarios and, therefore, widely used for training and testing anomaly detection, threat hunting, and incident response in as near-real conditions as possible. The structure of the data can often be designed to represent a variety of attack vectors, such as phishing, malware, and lateral movement [5].

3.4. ESX-1 & ESX-2 Dataset

The datasets ESX-1 and ESX-2 emanate from experimental environments devoted to virtualization security and cloud computing. They will be helpful in the assessment of hypervisors and virtualized infrastructure security. In these datasets, one may find information on attempted hypervisor attacks, exploitation of virtual machine escapes, and violations of data integrity. They are used in researching and benchmarking algorithms designed to protect virtualized environments against modern threats, which have provided a mainstay in research into cloud security [5].

3.5. Kyoto-Honey Dataset

Data collected from the honeypot system deployed in the network environment of Kyoto University comprise the Kyoto-honeypot dataset. It captures in detail the activities of the malicious actions such as scanning, exploiting, and brute-forcing attempts. The data includes features like timestamp, source IP, destination IP, service port, protocol, and types of attack. This dataset is well-known for investigating patterns of attacks and trends, creating IDS models, and training of machine learning algorithms used for threat detection in real-world environments [5].

3.6. USNW-NB15 Dataset

The UNSW-NB15 dataset is a voting-based benchmark dataset for network intrusion detection meant to alleviate the shortcomings of NSL-KDD and other datasets. With a realistic synthetic traffic model, the dataset is able to imitate both normal and attack behaviour using such features as IP headers, payloads, and protocols. Attacks are classified into nine categories such as DoS, exploits, and reconnaissance, allowing for a balanced, varied, yet rich collection of training examples. It is widely used by researchers to test machine learning model performances in

detection and prediction of modern cyber threats which have become sophisticated with further intelligence [3][4][13].

3.7. CIU Dataset

Data from Cyber Intrusion University is a stimulated dataset designed with the purpose of modelling various cyberattacks and normal network traffic. It includes logs and metadata derived from a variety of sources including web applications, firewalls, and intrusion detection systems, to simulate a real-world network environment. The dataset is thus most appropriate for training and evaluating intrusion detection systems as noted in both academic and experimental research. Being able to label different kinds of attacks and subsequently diverse kinds of traffic types makes the dataset an invaluable resource for cyber security studies, especially in respect of modern threats [4][13].

3.8. CMU CERT Dataset

The CMU CERT dataset is designed for insider threat detection research and generated by the CERT Division at Carnegie Mellon University. It represents a simulation of actual organizational activities and consists of both normal and malicious insider behaviours. It contains network logs, user activities, email exchanges, and other operational data. It serves the purpose of studying the four insider threat dynamics: data exfiltration, unauthorized access, and privilege escalation. The dataset also forms an essential base for building machine learning models to counteract insider threats [1].

3.9. Network Intrusion Detection Dataset

In fact, a collection of such labelled instances of network traffic is now widely available, spanning normal and malicious activities, including scanning, denial of service, and attempts to escalate privileges. Examples of such datasets include NSL-KDD and CICIDS2017. They are critical to training and testing machine learning algorithms used to detect and prevent network intrusion. Each characterizes various attack scenarios and asymmetries in traffic patterns providing a sturdy baseline for interns since intrusion detection models will be benchmarked both in simulated and confirmed environments [6].

3.10. Android Malware Detection (DERBIN-215) Dataset

Derbin-215 is a specialized dataset of Android applications, a monstrosity of both benign and malicious examples. Attributes like permissions, API calls, and network behaviour may be derived from Derbin-215, providing a good insight for malware detection. This dataset is used by researchers to develop and evaluate both static and dynamic detection tools for attacks against Android-based ecosystems. With a specific focus on mobile security, Derbin-215 plays an important role in the progress of machine learning models and the crafting of cybersecurity solutions against Android-targeting threats [6].

3.11. IoT Cyber Threat Detection Dataset

The dataset IoT Cyber Threat Detection registers the traffic and attack data specifically targeting IoT devices. Attack types endure within aspects of botnets, weak authentication exploits, and device misconfigurations, which are representative of frequent IoT vulnerabilities. The dataset is crucial in training and testing intrusion detection systems customized for IoT contexts, consequently working towards securing devices for smart homes, industrial systems, and healthcare applications. It sets the stage for real-life investigations of threats that provides further inputs towards the development of resilient cybersecurity solutions within the challenges that continue to grow within IoT security [6].

3.12. DARPA & ISCX Datasets

The DARPA and ISCX datasets form established benchmarks in cybersecurity research. Conceived by MIT Lincoln Laboratory, the DARPA dataset holds an important place as one of the earliest comprehensive traffic datasets aimed at the evaluation of intrusion detection systems (IDSs). This dataset contains alternatives of simulated traffic with attack scenarios such as denial of service (DoS), probing, and remote access attacks. The ISCX dataset, developed by the Information Security Center of Excellence, reflects modern network traffic patterns and attack types. Both these datasets are quite useful in evaluating IDS models and further refining the robustness of machine learning implementations in detecting intrusion into networks in an effective manner [2][13].

4. Results

Papers	Dataset Used	Model Architecture	Accuracy
[12]	NSL-KDD dataset	LSTM-RNN	0.884
[10]	NSL-KDD, CICIDS2017, Two additional datasets	Modified ANN	92%
[6]	Network Intrusion Detection Dataset	Machine Learning Models	Not specified
	IoT Cyber Threat Detection Dataset	GBM)	0.9856
	Android Malware Detection Dataset (Drebin-215)	(GBM)	0.9812
[2]	KDD Cup 99	Random Forest	Not specified
	NSL-KDD	(SVM)	92%
	CICIDS 2017	(CNN)	95%
	UNSW-NB15	(RNN), specifically LSTM	93%
	Combined contemporary datasets	Hybrid AI	90%
[5]	NSL-KDD dataset	EP-FCNN	0.958
	CICIDS 2017	EP-FCNN	0.98
	ESX-1 dataset	EP-FCNN	0.933
	ESX-2 dataset	EP-FCNN	0.947
	NSL-KDD dataset	EP-CNN	0.952
	CICIDS 2017	EP-CNN	0.98
	ESX-1 dataset	EP-CNN	0.952
	ESX-2 dataset	EP-CNN	0.936
	ESX-1 dataset	EP-LSTM	0.923
	ESX-2 dataset	EP-LSTM	0.947
	NSL-KDD dataset	Conventional Machine Learning Models	0.941
	EP-FCNN	Conventional Machine Learning Models	0.819
	ESX-1 dataset	Conventional Machine Learning Models	0.90
	ESX-2 dataset	Conventional Machine Learning Models	0.85
[7]	CSIC-2010, CICIDS-2017	CNN – LSTM (Hybrid Model)	91-93%
[3]	NSL-KDD	Stacked Autoencoder with Softmax Classifier	99.99%
	UNSW-NB15	Long Short-Term Memory (LSTM)	98.8%
	KDD99	Decision Tree (e.g., IntruDTree)	98%
[4]	NSL-KDD	RBF-SVM (Radial Basis Function – SVM)	99.90%
	KDD Cup 99	DNN	97.79%
	UNSW-NB15, Kyoto-Honeypot Dataset	Seq2Seq	99.90%
	CTU Dataset	DBN-based model	69.77%
	Not specified	DGA domain detection	96.43% (avg), 97.79% (highest)

[13]	NSL-KDD	(CNN)	96%
	UNSW-NB15	(LSTM) Networks	95.3%
	KDD Cup 99	(DBN)	98.6%
[11]	NSL-KDD	Autoencoder for feature reconstruction and anomaly detection	80%
	NSL-KDD	VAE for feature extraction + DNN for classification	89.08%
	Test-bed environment (custom dataset)	LSTM for time-series anomaly detection	98%
	Test-bed environment	DBN for detecting malicious traffic in IoT networks	97%
	CSE-CIC-IDS2018	CNN for classifying IoT-based intrusion traffic	99%
	N-BaIoT dataset	CNN for spatial feature extraction + LSTM for temporal analysis	94.30%
	KDDCup99	Deep Metric Learning for robust feature representation	99.78%
[8]	ISCX 2012 dataset	Stacked Autoencoder (SAE) + Support Vector Machine (SVM)	High accuracy, not specified
	KDDCup99 dataset	Deep Autoencoder (DAE) with a softmax classifier	Improved detection rate, not specified
	NSL-KDD dataset	Restricted Boltzmann Machine (RBM) + SVM	High accuracy, improvements over DBN models
	UNSW-NB15 dataset	Deep Belief Network (DBN)	Improved accuracy, detection rate, and F1-score
	UNSW-NB15 dataset	RNN + LSTM	High detection rate, low false-positive rate
	NSL-KDD dataset	Deep Neural Network (DNN)	Good accuracy for multiclass classification
	KDDCup99 dataset	(CNN)	High accuracy, better than traditional methods
[9]	Malling dataset	CNN + LSTM	96.3%
	Ember dataset	CNN	High accuracy, specific percentage not provided
	Ember dataset	LSTM layers	Improved performance, specific percentage not provided
	Dataset 1 & 2 from VirtualBox environments	FCNN	Good detection, specific percentage not provided
	Privately collected samples	CNN + LSTM layers	High accuracy, outperforms classical methods

5. Discussions

5.1. Overcoming Intrusion Detection Systems Challenges Using AI and Machine Learning

Effective Intrusion Detection:
These are promising detection capabilities provided by AI-based models, such as LSTM and RNNs, for complex attacks like U2R and R2L that were largely missed by traditional methods. The network events in a sequence are processed to identify the pattern of such complex attacks. For instance, the RNN-based models leverage the ability to learn temporal dependencies for better generalization on evolving attack patterns. Reference publications include works by [12][1] and [7].

Temporal and Feature Extraction:

Deep learning architectures like CNNs, autoencoders, and RBMs take away the pain of feature extraction from raw network data. This then does not require domain expertise for handcrafting features. Temporal models like LSTMs allow systems to pick up attacks that are buried in time-series data through learning patterns in sequences [12][11][8].

Real-Time Analysis:

The lightweight versions of CNN and RNN allow large-scale traffic processing in real time. These models are fine-tuned for domains such as SCADA systems and IoT environments, which guarantee a response to time-critical threats with high detection accuracy of anomalies [7][11].

Data Challenges:

Advanced techniques involve random weight allocation, federated learning, and data augmentation, solving problems of data imbalance, noisy datasets, and scarcity of labelled data. For example, in the case of intrusion detection, federated learning frameworks such as DeepFed have the systems act on decentralized data in a privacy-preserving manner [10] [8][9].

Integration and Scalability:

The AI systems are scalable, where they can integrate with distributed platforms like Apache Spark and use federated learning to handle scalability in large networks. These systems enable intrusion detection in real time for distributed environments like IoT by load balancing computation on several nodes [5][9].

Improved Accuracy and Reduction in False Positives:

For instance, models such as LSTM-IDS and CNN-IDS routinely achieve accuracy rates above 99% in the detection of common attack types, including Denial of Service and Probing. Moreover, an advanced tuning of parameters together with the use of denoising layers in autoencoders reduces false alarms and makes those systems more dependable under practical conditions [1][4][8].

Zero-Day Attack Detection:

The AI-based systems use anomaly-based techniques like autoencoders, CNNs, and hybrid methods to detect unseen-zero-day-attacks. Their working mechanisms are centred around analysing deviations in the normal traffic

pattern; hence, these are able to find novel threats without knowledge of the signature. This they do through the following works: [7][9].

Robustness against Adversarial Attacks:

Enhancements in model design, such as adversarial training and integration of multi-view data, improve intrusion detection systems' resilience against adversarial manipulations regarding static, dynamic, and image-based information. Thus, it guarantees better stability and reliability when facing evolving threats [10][6].

5.2. Challenges Not Fully Overcome in Intrusion Detection Systems Using AI and Machine Learning

Limitations of Dataset:

Most of such research are based on very old datasets like NSL-KDD, KDD Cup 1999, etc., which cannot grasp modern-day network traffic and attack scenarios. This leads to model over-fitting in controlled environments while actual deployment models remain unsuccessful. Newer datasets like UNSW-NB15 represent slight improvements but are nonetheless limited in scope [1][11][8].

High Computational Costs:

These models require vast amounts of computational power for training and deployment, so it is pretty challenging to utilize such complex architecture models, including CNN-LSTM, in resource-constrained settings like IoT devices at the edge with low power and processing capabilities [12][3][11].

False Positives and Alarms:

Despite these advances, most IDSs continue generating high false positive rates when deployed in the real world. These false alarms create a significant overhead for the SOCs through manual review processes and desensitize analysts to real threats [7][11].

Dynamic Threat Response

It is important to realize that the nature of cyber threats is inconstant evolution; thus, IDSs should also be updated regularly. Models very often cannot keep up with such dynamic evolutions, especially when traffic is encrypted and methods of attack are sophisticated-such as in APTs. This is while most of the existing systems lack robust mechanisms with regard to continuous update and retraining in order to maintain their accuracy in dynamic environments [13][6][9].

Limited scope and methodological gaps

Unfortunately, many such studies have a focus on restricted attack types or datasets; thus, most of them have limited generalizability. The diversity of validation procedures is incomplete, and the discussion about cross-domain adaptation is seldom found, such as transfer learning. This seriously restricts practical applicability in diverse real-world scenarios [4][10][8].

Energy Efficiency:

Deep learning models do need computationally intensive GPUs or cloud computing, hence they are unsuitable for environments that have energy supply constraints, such as IoT devices. Research into alternatives with low power consumption, such as FPGA or reservoir computing, has remained inadequate [3][11].

Adversarial Vulnerability:

Even with these advances, deep learning models are still vulnerable to adversarial attacks, wherein an attacker crafts certain inputs to intentionally fool the IDS. While a few research works have suggested partial solutions, general defences against adversarial samples are still absent [10][6][8].

Integration Challenges:
Incorporating AI models into diverse enterprise systems poses significant challenges, including compatibility with existing infrastructure, managing heterogeneous data, and ensuring seamless deployment. These issues reduce the practical effectiveness of AI-powered cybersecurity solutions [13][6][9].

6. References

[1] F. R. Alzaabi and A. Mehmood, "A Review of Recent Advances, Challenges, and Opportunities in Malicious Insider Threat Detection Using Machine Learning Methods," *IEEE Access*, vol. 12, no. February, pp. 30907–30927, 2024, doi: 10.1109/ACCESS.2024.3369906.

[2] I. H. Sarker, M. H. Furhad, and R. Nowrozy, "AI-Driven Cybersecurity: An Overview, Security Intelligence Modeling and Research Directions," *SN Comput. Sci.*, vol. 2, no. 3, pp. 1–18, 2021, doi: 10.1007/s42979-021-00557-0.

[3] I. H. Sarker, "Deep Cybersecurity: A Comprehensive Overview from Neural Network and Deep Learning Perspective," *SN Comput. Sci.*, vol. 2, no. 3, pp. 1–16, 2021, doi: 10.1007/s42979-021-00535-6.

[4] H. Chaudhary, A. Detroja, P. Prajapati, and P. Shah, "A review of various challenges in cybersecurity using artificial intelligence," *Proc. 3rd Int. Conf. Intell. Sustain. Syst. ICISS 2020*, pp. 829–836, 2020, doi: 10.1109/ICISS49785.2020.9316003.

[5] J. Lee, J. Kim, I. Kim, and K. Han, "Cyber Threat Detection Based on Artificial Neural Networks Using Event Profiles," *IEEE Access*, vol. 7, pp. 165607–165626, 2019, doi: 10.1109/ACCESS.2019.2953095.

[6] M. Schmitt, "Securing the digital world: Protecting smart infrastructures and digital industries with artificial intelligence (AI)-enabled malware and intrusion detection," *J. Ind. Inf. Integr.*, vol. 36, no. September, p. 100520, 2023, doi: 10.1016/j.jii.2023.100520.

[7] A. Kim, M. Park, and D. H. Lee, "AI-IDS: Application of Deep Learning to Real-Time Web Intrusion Detection," *IEEE Access*, vol. 8, pp. 70245–70261, 2020, doi: 10.1109/ACCESS.2020.2986882.

[8] J. Lansky *et al.*, "Deep Learning-Based Intrusion Detection Systems: A Systematic Review," *IEEE Access*, vol. 9, pp. 101574–101599, 2021, doi: 10.1109/ACCESS.2021.3097247.

[9] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, and S. Venkatraman, "Robust Intelligent Malware Detection Using Deep Learning," *IEEE Access*, vol. 7, pp. 46717–46738, 2019, doi: 10.1109/ACCESS.2019.2906934.

[10] T. S. Oyinloye, M. O. Arowolo, and R. Prasad, "Enhancing Cyber Threat Detection with an Improved Artificial Neural Network Model," *Data Sci. Manag.*, 2024, doi: 10.1016/j.dsm.2024.05.002.

[11] M. A. Alsoufi *et al.*, "Anomaly-based intrusion detection systems in iot using deep learning: A systematic literature review," *Appl. Sci.*, vol. 11, no. 18, 2021, doi: 10.3390/app11188383.

[12] M. Ibrahim and R. Elhafiz, "Modeling an intrusion detection using recurrent neural networks," *J. Eng. Res.*, vol. 11, no. 1, p. 100013, 2023, doi: 10.1016/j.jer.2023.100013.

[13] C. S. Ravi *et al.*, "AI-Powered Intrusion Detection Systems : Real-World Performance Analysis," vol. 4, no. 1.