Journal of Positive School Psychology http://journalppw.com
2022, Vol. 6, No. 4, 473-485

Design Machine Learning BasedIntelligent Techniques for Detecting Network Attacks

Shekjavid Hussain[1]; Dr. Bechoo Lal[2]

[1]Research Scholar, Department of Computer science & Engineering; Shri JJT University Jhunjhunu Rajasthan

[2]Assistant Professor, Department of Computer science & Engineering; Shri JJT University Jhunjhunu Rajasthan

Abstract

The number of Internet of Things (IoT) devices that are vulnerable to cyber-attacks is increasing at an alarming rate. As a result, network operators are placing an increasing emphasis on the control of these devices. A comprehensive packet inspection in software can be difficult, expensive, rigid, and unable to scale with current network monitoring solutions that use specialised acceleration on network switches. SDN and machine learning are used in this work to take use of the programmability offered by SDNs.

Information driven models for overseeing IoT gadgets in light of their organization exercises by means of stream based telemetry. The three manners by which we have an effect: Over a six-month time frame, we gathered traffic follows from 17 genuine purchaser IoT gadgets and recognized a bunch of traffic streams (per-gadget) that portray the organization conduct of different IoT gadget types and their working states (i.e., booting, effectively collaborated with client, or being inactive). (2) We create a multi-stage design of surmising models that utilization stream levity information to make forecasts about the organization conduct of different IoT gadget types and their working states. (3) We measure the compromise among execution and cost of our methodology and clarify how our checking framework can be used in activity to identify conduct changes, all utilizing genuine traffic information to prepare our models (firmware overhaul or digital assaults)..

Keywords: WSN, IoT, Cyber-attack, Security, Machine learning,

1. Introduction

Machine learning (ML) and data mining (DM) methodologies for cyber security applications were surveyed in this article. As well as many applications to cyber intrusion detection challenges, the ML/DM approaches and techniques are explained. Paper discusses the difficulty of ML/DM algorithms in terms of complexity and recommends which strategies to utilise depending on what kind of cyber problem you are trying to address.

When it comes to protecting computers against assault, unauthorised access or modification or even destruction, cyber security encompasses a wide range of technology as well as processes. There are two main types of cyber security systems: network and computer (host). Firewalls, antivirus software, and intrusion detection systems are all included at the very least in each of these systems (IDS). Unauthorized use, copying, change, and destruction of information systems can be discovered, determined, and identified with the aid of IDSs [1]. Internal and external invasions are two types of security breaches that have occurred (attacks from within the organization).

Cyber analytics that enable IDSs fall into three broad categories: misuse-based (also known as signature-based), anomaly-based, and hybrid. By looking for patterns in previously detected assaults, misuse-based techniques can identify potential threats before they can be launched. There is no need to worry about a flood of false alarms because they are excellent at detecting established assaults. There are rules and signatures in the database that need to be

© 2021 JPPW. All rights reserved

updated frequently. Novel (zero-day) threats cannot be detected with misuse-based approaches. It is possible to identify network and system anomalies by using anomaly-based techniques that simulate normal network and system behaviour. This is because they can identify zero-day assaults, which is why they're appealing. A further benefit is that the profiles of typical activity are tailored to each system, application, or network, making it more difficult for attackers to figure out what they can do undetected. It is also possible to create signatures for abuse detectors using data that anomaly-based approaches (novel attacks) identify. While anomaly-based approaches have the advantage of detecting previously unknown (but valid) system behaviours, they also have the drawback of increasing false alarm rates (FARs). Misuse and anomaly detection can be detected using a combination of hybrid techniques. For known incursions, they increase detection rates and reduce false positive (FP) rates.

The majority of anomaly detection methods found in a comprehensive analysis of the literature were hybrid in nature, rather than being pure. Because of this, the methods for detecting anomalies and creating hybrid models are discussed simultaneously in ML and DM textbooks.

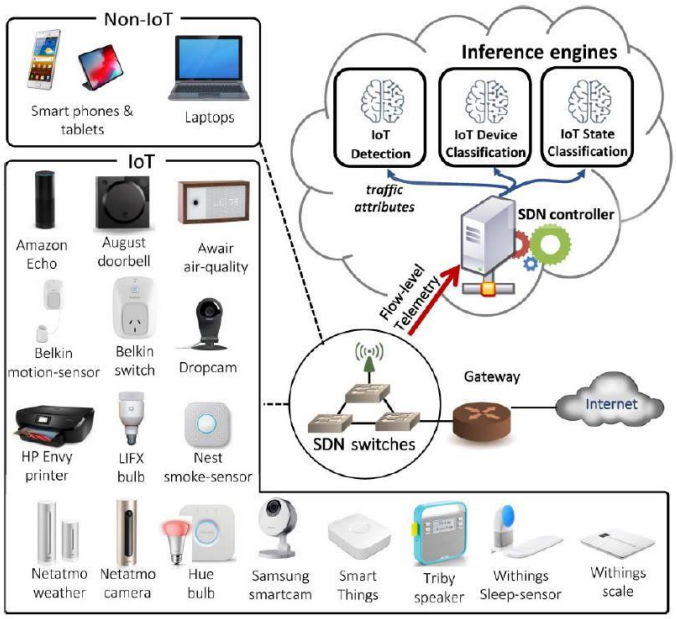

Fig. 1. System architecture of network telemetry and inference engines.

Network-based and have based interruption recognition frameworks (IDSs) are two other subcategories of IDSs. An interruption recognition framework (IDS) in view of an organization screens network traffic to identify interruptions. The product climate associated with a particular host is checked by a host-based IDS, which watches interaction and document movement.

Focused on machine learning and data mining (ML/DM) techniques for cyber security, this research focuses on the ML/DM methodologies and their descriptions. Several evaluations of these procedures have been published, as well as numerous papers explaining these techniques. Unlike past studies, our paper focuses on publications that meet a set of predetermined standards. Using "machine learning" and cyber, as well as "data mining," we ran Google Scholar queries. Due to the fact that they presented widely used approaches, highly cited papers received extra attention. In order to avoid omitting important new and emerging techniques, several publications on these topics were included as well. At the end of the selection process, papers were chosen to represent each of the ML/DM categories listed below.

To anyone interested in learning more about machine learning and data mining in the context of cyber intrusion detection, this paper is a good place to start. As a result, much focus is placed

© 2021 JPPW. All rights reserved

on providing a full discussion of the ML/DM approaches, as well as citations to foundational publications for each method.. A few examples of how the concepts were put to use in cyber security are given.

2. OVERVIEW OF MACHINE LEARNING

As part of the debate on machine learning in CPS and the importance of making ML models resistant to adversarial attacks, the various ML models widely used in CPS are briefly described in this section.

According to a popular ML definition, machines can make intelligent decisions without being explicitly programmed. People often confuse machine learning with artificial intelligence (AI), although in reality, the two are distinct fields. Approaches to machine learning that are based on data are known as ML. As a result of the massive amounts of data collected by the multiple sensors, machine learning is being used in CPS. As a result, ML approaches are typically divided into three categories: supervision, unsupervised, and reinforcement learning (RL). Figure 3 depicts the various types of work and the associated duties. In this part, we'll take a look at these issues. Each of these classes will be briefly discussed, as well as some of the algorithms associated with them. In many ways, this section lays the groundwork for the article's primary focus.

A. Supervised Learning

Unsupervised learning uses data samples and a label for a desired answer or solution (s). Because the ML method is designed to create a function that maps input to output, its purpose is to create a mapping. An efficient model may take an unknown input and determine what the output should be once learning has taken place. Accuracy, precision, recall, and F1-score are some of the most often used metrics for ML systems. In this category, classification and regression are the most important tasks. Classification tasks are the primary focus of supervised learning in the context of CPS. In the following, we'll take a look at some of the most popular CPS algorithms.

There are two types of artificial neural networks (ANN): biological neural networks (BN) and artificial neural networks (ANN). One of the simplest ANN structures is the perceptron. A preset threshold can be used to train perceptrons to make predictions. The MLP (multi-layer perceptron) delivers superior outcomes by integrating multiple perceptrons. Input signals are changed into yield announces initiation capacities like paired advance, sigmoid, and redressed straight unit ReLU. They utilize numerical calculations to decide whether a neuron ought to be enacted.

ANN models have been utilised to solve a wide range of issues in modern society. DNNs, the subject of a subsequent section, have emerged as a go-to model for a variety of classification and regression problems in CPS research.

Classification, regression, and even outlier identification can all be accomplished using SVM thanks to the model's accuracy and efficiency, making it a go-to ML model in CPS research. 2) Support Vector Machine (SVM) An N-dimensional hyperplane (decision boundaries) is sought by the algorithm in order to categorise the data points. The hyperplane's size is proportional to the dataset's total number of features. Planes should be chosen that maximise data points between the two classes. SVM was a common supervised learning ML method before neural networks became widely utilised.

To classify data, k-Nearest Neighbors (kNN) uses machine learning to find the closest neighbours in a set of points in space. Data mining and intrusion detection are only some of the uses of pattern recognition. Because of its non-parametric nature, there is no need to make assumptions about the data needed for its use in real-life applications. According to the algorithm, the class of a test point is determined by the majority of its K nearest neighbours. KNN uses the training data directly to make predictions. In order to make predictions about a new instance, one can search over the entire dataset in search of the K closest examples or neighbours, and then sum the output variable for each of those K cases. Distance measurement methods like these are used to determine how similar two occurrences are.

Distance in terms of the Euclidean geometry. Even while kNN isn't as widely used as it once was, some researchers are still utilising it in their work.

B. Unsupervised Learning

© 2021 JPPW. All rights reserved

Unsupervised learning, on the other hand, relies on unlabeled training data, as opposed to the supervised learning approaches outlined above. Data that hasn't been tagged is the focus of the algorithm's investigation. This type of ML is used to do tasks including dimensionality reduction, clustering, density estimation, anomaly detection, and visualisation. In the following, we'll go through two of the most often utilised algorithms in CPS research.

One of the simplest and most widely used unsupervised machine learning algorithms is K-Means clustering. Clustering is the process of discovering and grouping together instances that are similar, so that patterns can be discovered. Simply said, the goal of K-means clustering is to divide a set of data points into K clusters. K number of centroids (the centre of the cluster) are identified and each data point is assigned to the nearest cluster, with the ultimate goal of minimising centroids. Even though this method is fast and scalable, it suffers from restrictions when the clusters have variable sizes and differing density. CPS applications, on the other hand, have employed it extensively to do data analysis and dimensionality reduction as well as anomaly detection and picture segmentation.

Using PCA, the nearest hyperplane to the data is found and the data are then projected onto it. To put it another way, the data is transformed into a new coordinate system via an orthogonal linear transformation. While PCA can reduce the amount of features in a dataset, it retains all the information necessary for training, making it an ideal tool to use in machine learning research. Therefore, PCA is employed in conjunction with other dimensionality reduction algorithms, such as linear discriminant analysis (LDA).

C. Reinforcement Learning (RL)

As the algorithm or agent interacts with the environment, it learns to make decisions. In a trial and error method, the algorithm learns by receiving rewards and penalties for correct and incorrect actions. As a result, the agent's ultimate objective is to obtain the greatest possible profit in any given situation. A RL system is depicted in Figure 4 with the agent and environment intertwined. the external conditions or objects that the agent is acting on are depicted by this environment. The policy, reward signal, value function, and environment model are all crucial components of an RL system. It is the policy that dictates how an agent acts at any particular moment in time. Map the states to the activities is the most common way to accomplish this. The major purpose of the setup is to provide the agent with a reward based on their current actions and the current state of their environment. As a general rule, the agent will adjust the policy in order to maximise the reward. Value functions, albeit similar to reward signals, depict the long-term or cumulative reward an agent can accrue depending on the states that are expected to follow the current state and the rewards associated with those future states. Based on data it possesses about a current state and action, an environmental model tries to derive predictions about the agent's next state and rewards.

The trade-off between exploration and exploitation distinguishes the RL from other learning algorithms. The agent's ultimate objective is to maximise the reward it receives from its interactions with the environment, which necessitates that it strive to exploit the information it has gained from previous interactions and the rewards it has received. However, the agent must investigate additional options in order to maximise benefits in the future while choosing better actions. The exploration-exploitation dilemma is the term used to describe this situation.

RL algorithms have been developed over the years. Watkins and Dayan first proposed the Q-learning algorithm. Next, Google DeepMind's deep Q-Network (DQN) [33, 34] popularised the DRL concept in 2013. Many others have been proposed, such as the value iterative network (VIN) [35], asynchronous advantage actor-critic algorithm (A3C) [36], trust region policy optimization (TRPO) [37] and the unsupervised reinforcement and auxiliary learning (UNREAL) [40]. These are just some of the many approaches that have been put forth in the past. Furthermore, Google DeepMind's DQN, A3C, and UNREAL have had a significant impact on research in RL, and it is important to point this out. Following sections will show that DQN-based applications of RL in real-world contexts are prevalent. There has been some research into the defence of RL algorithms, such as the DQN

© 2021 JPPW. All rights reserved

and TRPO. Two of the most often used RL algorithms in CPS are described here.

One of the most frequently utilized RL calculations is Q-Learning. A specialist's main role is to become familiar with the Q-Value through its connections with the climate, then, at that point, use that data to make the legitimate move. To decide the most ideal state-activity esteems, the Q-Value is utilized. To lay it out plainly, the Q-Value alludes to the limited collected compensations of a specialist that starts with a state-activity pair and sticks to explicit arrangements. At each given time, the's specialist will likely make the move with the most noteworthy Q-Value. The Q-Value is first assessed to nothing and afterward refreshed utilizing the QValue emphasis process. Q-disappointment figuring out how's proportional well to large Markov choice cycles with many states and activities is an issue that additionally hampers its use in CPS.

2) Deep Q Network (DQN): DNNs are frequently used to estimate Q-Values in order to overcome the aforementioned scalability difficulty of Q-learning. Accordingly, DRL was raised by the DQN. With these four systems, DQN can conquer the troubles of temperamental learning: experience replay, target organization, reward cutting, and casing skipping. The DQN is prepared utilizing state change tests established by communications with the climate that are put away in a replay memory. An objective DQN is likewise used to produce target esteems. The DQN algorithm's exceptional performance has helped it rise to prominence in Research conducted by the Center for Psychological Science. When it comes to driverless vehicles, this is especially true.

In scenarios where an active decision-making agent interacts with its environment, RL reflects circumstances in which the agent strives to effectively attain a goal in the environment despite its ignorance of the environment. To put it another way, it relies on its ability to influence how things will turn out over time, and hence how many options will be accessible to it.

D. Deep Learning

Data science has become a major focus of recent research, particularly in the application of DL. Deep learning (DL) approaches differ from standard shallow algorithms because they have numerous hidden layers, conduct high level feature abstraction, generalise better on unseen samples, and have proved to increase the performance of systems in which they have been applied. DL's unique features have made them a popular choice for a variety of CPS jobs. In the context of CPS, DL techniques like as convolutional neural networks, recurrent neural networks, and autoencoders have been applied.

3. Related Work

There is a lot of interest in network traffic measurement in academics and industry. Traditional port-based counting with SNMP [25] and packet sampling [24] have been proposed and practically deployed, as have flow-based telemetry [21], [22], and WiFi packet sniffing [26], [27].

Modern telemetry approaches can be divided into two categories: (a) packet-based and (b) flow-based. Randomly sampling (i.e., one in N) packets from network switches is one of the most often used sFlow methods. Random sampling means that packets from elephant flows (traffic-heavy andlong-lasting) are more likely to be collected, resulting in incorrect measurements. According to Everflow [28], a solution to this problem is to use the match and mirror capability of data centre switches to gather certain packets (e.g. TCP SYN, FIN, and RST). In Planck [29], data from several ports is mirrored to a monitoring port where a collector uses high-rate sampling to evaluate flow throughput at very short timescales. As a general rule, packet-level telemetry can only provide a limited view of network traffic.

Flow records are exported by commercial switches equipped with NetFlow [22] engines (IPFIX). It is possible to export IPFIX records containing a wide range of information [20] from the network traffic via a Netflow-capable switch. There are, however, two fundamental drawbacks:

When a flow record expires, it is not exported immediately, and the switch's computational costs are considerable for updating and maintaining flow records. Data structures for flow counters (encoded hash tables) with low memory overheads and periodic exports of flows make FlowRadar [21] more efficient than Netflow (e.g., 10 ms). There are, however, no commercial switches that support FlowRadar.

© 2021 JPPW. All rights reserved

With the help of SDN APIs [30], we can measure traffic flows at minimal cost and with suitable resolutions in this study.

Various applications, including network management [32], quality of service [33], and cyber-security [34], make extensive use of traffic classification. IoT traffic classification has gotten academics' interest recently [35] because it can help them identify IoT devices, as well as their current states and any anomalous activity. The organization traffic of Internet of Things (IoT) gadgets has likewise been utilized in a few investigations [36], [37]. [38] endeavors to connect network exercises of Nest Thermostat and Nest smoke-sensor to client action by dissecting the conveyance of payload sizes for different organization streams. [39] shows that specific IoT gadgets (behind NAT) can be perceived by the pace of traffic to determined Internet endpoints. They don't naturally recognize or characterize Internet of Things gadgets.

North of 300 traits (parcel level and stream level) of IoT traffic are utilized to prepare an AI model in [40]. There are other key measurements to consider, for example, parcels Time-To-Live (least, middle, and normal), how much bundles with a reset banner, and the Alexa rank for servers the gadget is in correspondence with.. Utilizing 16 double highlights (addressing utilization of different conventions at application, transport, organization and connection layers) joined with distant IP address/port numbers, bundle size and crude byte worth of IoT traffic, a managed multi-class classifier was prepared.

Gadget arrangement is great, yet the expense of characteristic extraction is high since bundle investigations are required. One more methodology in [42] presents a structure for classifying gadgets as indicated by their semantic kind (for instance, camera, wellness/clinical gadget or climate sensor). It is feasible to make a model with wide limits by gathering gadgets of a similar sort (e.g., cameras from various makers) in light of the fact that these gadgets regularly contrast in their organization conduct. Accordingly, during the testing stage, expansive models would deliver a high pace of erroneous order. Rather, in this review, we centeraround a solitary IoT gadget for each class, expanding the precision of characterization and the responsiveness of models to conduct changes (e.g., camera of a particular producer). [43] affirms that machines might be prepared to identify inconsistencies in IoT traffic brought about by DDoS attacks using measurements like as bundle size, between parcel delay, normal data transmission and the quantity of unmistakable IP addresses identified for a brief timeframe with regards to network safety (i.e., 10-seconds). Volumetric assaults on IoT gadgets are identified by a machine created in [44] that screens stream rules produced from the MUD profile. Notwithstanding, we utilize assault traffic made by the creators of [44] to exhibit how our checking motors empower network administrators to accomplish more investigation into inconsistency (and assault) discovery (physically or by different frameworks).

Machine and profound learning headways have ignited a whirlwind of new review into IDS, with scientists using these procedures to create special IDS in the earlier ten years. There are a few studies that give a decent outline of the present status of the IDS discipline. They prescribe the work of unaided calculations to defeat the requirements of existing reasonable datasets and to find undetected, zero-day attacks. This review and others like it have observed empowering results with the utilization of autoencoders and other irregularity location procedures such detachment timberlands, one-class SVM, and head parts examination. Otoum et al. [21] as of late introduced a mixture IDS that utilizes a common mark based IDS to identify known assaults and an inconsistency based IDS to distinguish surprising attacks. Stream based order is utilized in the greater part of the proposed AI IDS. They are unaffected by encryption conventions since they are just made from the headers of bundles [22]. Traffic stream level elements are extricated as well as built utilizing four open source traffic stream analyzers by Khatouni et al. [23]. They had the option to recognize known administrations from a few encoded administration channels utilizing these stream highlights.

Collectively, these exploration show high characterization execution and propose their models for certifiable application, but functional

© 2021 JPPW. All rights reserved

utilization of these procedures is restricted. Following 10 years of exploration, little headway has been made in resolving this issue [24]. This was as of late recognized as an overall test in a review by a Google joint exertion and named under detail. In addition, Leevy et al. [25] have publicly questioned the remarkable high results published in the literature in the field of network intrusion detection. In addition, the study emphasises the significance of recording all procedures performed in order to reproduce. There are concerns about the generalisation performance of IDS in the actual world, and Ahmad et al. [26] ask for an effective approach to validate this in their future work. It has been shown in recent research that one way to reduce overfitting is to rank the features that are used and only select the highest forming ones, which has has the added benefit of decreasing computing complexity because the input dimension is much smaller. This is similar to Aloqaily et al. [28]'s D2H-IDS system, which employs a deep belief network to reduce the dimensions of an attack and a decision tree to classify it.

This study aims to overcome the gap between academic prototypes and practical implementations by presenting an alternative evaluation technique to estimate the generalisation strength of suggested models in the recent literature and laying the groundwork for future research in this area.

In order to train and evaluate machine learning-based IDS, a lot of data is needed. This data should ideally be taken from the real world and be a good representation of what is to come in future inputs. This is a difficult problem to solve since network traffic contains private information that should be protected. To get around this obstacle, you can anonymize the data. This way, no personal information can be linked to it. Operations like data aggregation and the removal and modification of particular attributes are common. This has been shown to be a time-consuming endeavour in the past. Netflix

There are many examples of how it went awry, such as the Prize. In 2006, the largest streaming service in the world made a dataset of movie evaluations from 500,000 users available to the public. Within weeks, the anonymization method had already been disrupted, exposing the sensitive information of specific users [29]. By creating a synthetic dataset, you can get an accurate representation of real-world network traffic. An experiment can be built up using a wide range of ways to mimic the network and its actual users and their related issues rather than gathering data from the network and its actual users. This strategy only works if the approaches employed closely resemble benign conduct. For intrusion detection, only a few realistic datasets exist. For the Fifth Conference on Knowledge Discovery and Data Mining in 1999, the Defense Advanced Research Projects Agency (DARPA) generated the most extensively used dataset, KDDcup99 [30]. Tavallaee et al. [31] presented a new version of the dataset, NSL-KDD, ten years after it was first released, along with a detailed study pointing out the problems with the original dataset. However, more than two decades after the original dataset was amended, the network protocols and assaults it utilised are no longer representative of modern communication networks and have been shown to be flawed [32]. An early leader in creating realistic network intrusion detection datasets was Canada's Institute for Cybersecurity (CIC). An intrusion detection dataset for benchmarking must meet 11 requirements, according to a study by Sharafaldin et al [33]. All 11 conditions are met by the first dataset, CIC-IDS-2017, which collects real network traffic over a five-day period utilising several Internet protocol types, including HTTP(s), FTP, SSH, IMAP, and POP3 [34]. A B-Profile system mimicking human interaction generates all of the traffic in this collection. The CIC and the Communications Security Establishment published CSE-CIC-IDS-2018 a year later (CSE). This dataset used the same techniques as CIC-IDS-2017, but on a much larger network within Amazon Web Services' cloud infrastructure (AWS). CICFlowMeter [35, 36] aggregates PCAP packets into bidirectional flows and distributes them as raw network packets (PCAP). Machine learning techniques make it simple to use in IDS.

4. ML-Assisted DDoS Attack Detection

With the use of stateful data planes and the P4 language, this study attempts to implement DAD in SDN networks using ML capabilities for

© 2021 JPPW. All rights reserved

effective attack detection by automatically obtaining traffic information (i.e., a signature). The two DAD designs, namely Standalone and Correlated DAD, are compared in terms of classification performance and algorithm complexity, including training and prediction times, as well as the impact of attack rates on the algorithms' performance.

4.1 *DAD* Detection Architectures

For an organization with five P4-empowered switches, we develop two particular DDoS assault location models: Standalone DAD and Correlated DAD, whose practical squares are displayed in Fig. 1a and b, separately. We describe the recognition of DDoS assaults in both Standalone and Correlated structures as a ML arrangement challenge. The discovery module yields a judgment for the noticed traffic, i.e., a name "1: assault" or "0: no-assault" to demonstrate whether or not an assault is available in the given time span, separately. Assuming various succeeding windows are classified as containing an assault, this mark can be utilized to settle on choices about bundle sending, for example, erasing parcels or sending picked parcels to the SDN regulator for more investigation. Honestly, our emphasis here is on the twofold arrangement of traffic windows with length T, rather than on explicit bundle sending choices. For instance, in the Standalone DAD plan, each P4 switch is outfitted with a DAD module that utilizations AI to distinguish DDoS assaults utilizing just privately recorded traffic. Interestingly, a single parent module gets traffic data from a few P4 switches and makes decisions in light of all around the world noticed traffic in the Correlated DAD engineering (see Fig. 1b). 1 (1) a highlights extractor and (2) an AI classifier contain the discovery module. Both Standalone and Correlated structures can be rearranged by re-appropriating specific undertakings (e.g., highlights extraction or even ML-based arrangement) straightforwardly to the P4 switches. Taking into account that data got from traffic streams is traded between identification module and P4 switches in different structures, for example, reflecting whole information parcels, their headers, or in any event, removing metadata (i.e., highlights) from a succession of information bundles, we likewise assess the extra inertness presented by the assault location module.

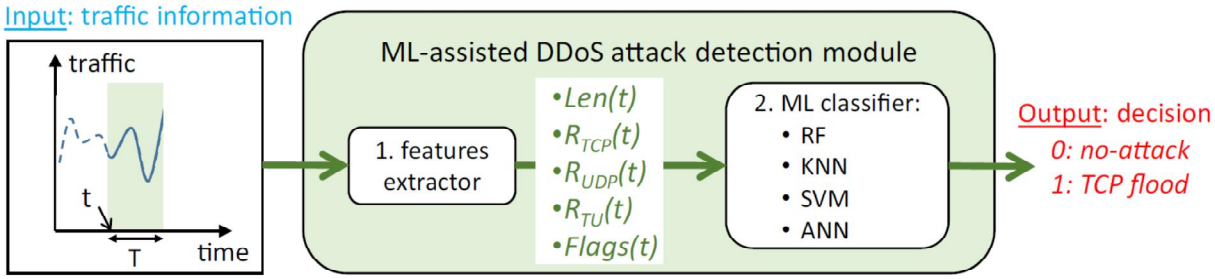

Fig. 2 Window features extraction and classification

5. Machine Learning Methods for Detecting Errors in the Internet of Things

On the NSL-KDD dataset, the suggested approach has been tested as a binary classification. The keras deep learning library for Python is used to implement this model. It is necessary to employ a CDNNIDS that has three fully connected input, hidden, and output layers. The pool's official dimensions are 2 by 2. To train the model, the three completely linked layers each have two neurons. This model has a dropout rate of 0.3. There are a number of indicators that are used to evaluate the proposed project

5.1 Data Set

The benchmark network traffic dataset NSL-KDD was used to evaluate the proposed PCRFE-CDNN-IDS system's performance. There is no better dataset for evaluating IDS than this one. A total of 41 qualities are grouped into three categories: basic, content-based, and time-based. 22 attacks make up the training set; 16 attacks constitute the testing set. First, there are denial-of-service assaults (DoS); second, there are probing attacks (PA) (PA) User-to-

© 2021 JPPW. All rights reserved

Root attacks and Remote-to-Local (R2L) attacks (U2R). Details about the IDS attacks and training/testing data are included in the binary class.

5.2 Features Selection

The Pearson Correlation based Recursive Feature Elimination feature selection strategy suggested for the NSL-KDD dataset eliminates unnecessary features from the feature set recursively and adds the selected features to the feature subset. It uses the dataset's 41 features. This proposed filter-based feature selection selects four important qualities for further processing. As shown in last section, alternative FS algorithms and the filter-based FS technique suggested here can both be used to compare their respective performance when it comes to feature selection. It shows the names of the features that were selected by the suggested model.

6. Proposed System Evaluation In terms of Feature Selection

Proposed PCRFE feature selection is used to evaluate this project's effectiveness in terms of both total number of features and features actually used. It shows the evaluation results. The table shows that by lowering the feature set, an accuracy of 99 percent can be achieved. First, the suggested CDNN-IDS with all of its features is tested, and it achieves a 91% accuracy rate. For classification of IDS data, using the proposed PCRFE Feature selection and deep learning model, the accuracy percentage of the classification was enhanced by 8 percent and reached a 99 percent accuracy rate. Figure 4 shows the results of this evaluation.

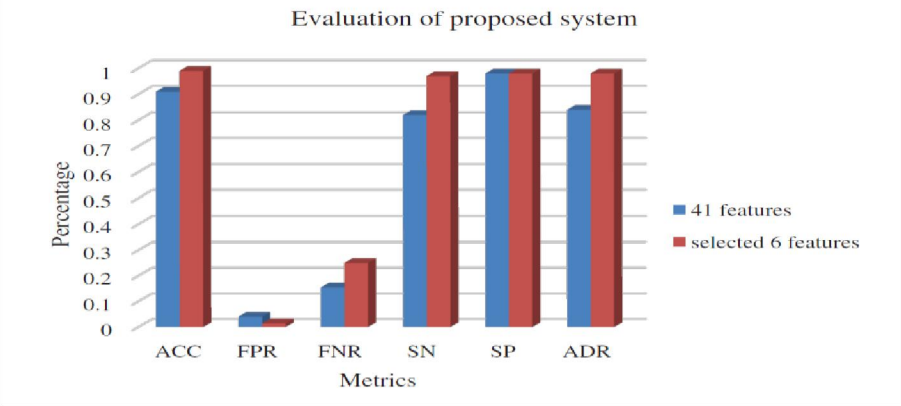

Figure 4: Evaluation of proposed system

For example, discrete differential equations [4], Gain ratios [5], symmetrical uncertainty [6], and ABCs are all compared to the suggested work feature selection performance. It shows the findings of the experiments, and Fig. 5 illustrates them. As a result of the evaluation, our findings Six features are proposed to be eliminated using the recursive Pearson correlation method. When compared to previous IDS feature selection techniques, this one achieved a classification accuracy of 99 percent. It is therefore possible to minimize the feature set and choose only those features that are essential for a reliable IDS system.

© 2021 JPPW. All rights reserved

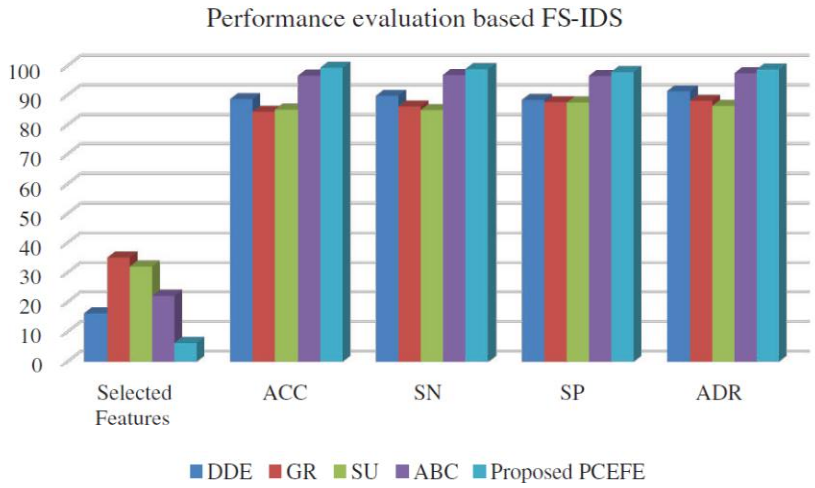

Figure 5: Illustration of different IDS feature selection with proposed FS

Performance Comparison of Proposed with Existing IDS Systems

A comparison of our proposed convolutional deep neural network-based IDS system with the existing IDS systems, such as DMNB [7], DBN-SVM [19], Bi-layer behavioral-based DMNB, TUIDS [32], FVBRM [33], PSOM (34), and LSSVM-IDS + FMIFS [2], is made. Tab. 6 displays the research findings. Figures 6 and 7 show the accuracy and FPR. As demonstrated in the experimentation, our suggested filter-based feature selection with deep learning IDS achieves a classification accuracy of 99.996% and the smallest false positive rate of 0.23. To put it another way, the suggested IDS is more accurate than others and has less false positive rates (FPR).

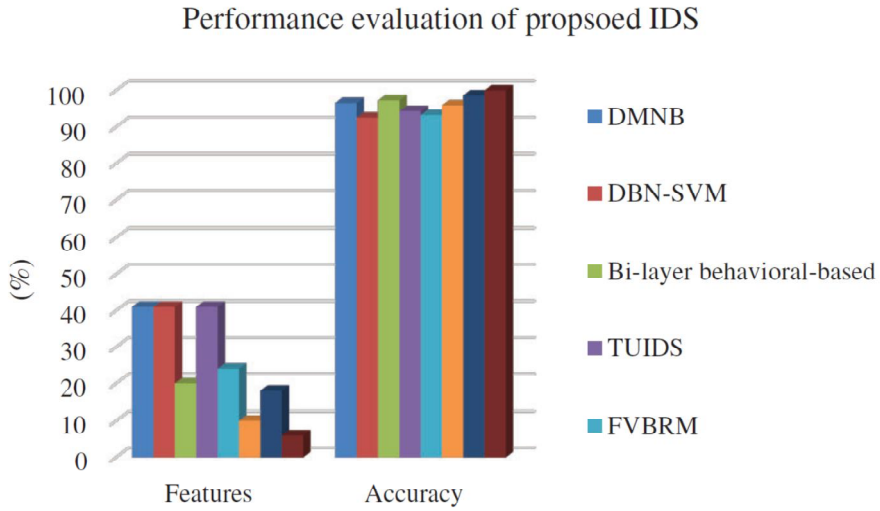

Figure 6: Illustration of the performance of proposed IDS

Recursion-based feature elimination, as demonstrated in this experiment with NSL-KDD dataset, lowers irrelevant characteristics and so increases the accuracy level. DoS, Probe, R2L, and U2R are all well-classified by our convolutional deep neural network. Our proposed deep learning technique to intrusion detection has been demonstrated to be more effective.

© 2021 JPPW. All rights reserved

7. Conclusion

PCRFE (Pearson correlation based recursive feature elimination) was presented in this research to reduce the redundancy among the features using recursive feature elimination and construct the relevant subset of features that are associated with Pearson correlation. In order to better detect intruders, a DL approach known as CDNN is used to classify the subset feature data. NSL-KDD dataset is used in the evaluation. The proposed PCRFE-CDNN-IDS has superior performance in detecting network intrusions based on the experimented results. Our proposed IDS may also be proven to be efficient by comparing it to other IDS. Multi-class classification with optimised feature selection strategies will be used in the future to increase the detection rate of the proposed IDS, and it will also be tested with other IDS datasets other than NSL-KDD in order to determine the efficiency of the proposed scheme.

8. References

[1]. L. Dhanabal and S. Shantharajah, "A study on NSL-KDD dataset for intrusion detection system based on classification algorithms," International Journal of Advanced Research in Computer and Communication Engineering, vol. 4, no. 6, pp. 446–452, 2015.

[2]. M. A. Ambusaidi, X. He, P. Nanda and Z. Tan, "Building an intrusion detection system using a filter-based feature selection algorithm," IEEE Transactions on Computers, vol. 65, no. 10, pp. 2986–2998, 2016..

[3]. Dalal, S. &Athavale, V. (2012). Analysing Supply Chain Strategy Using Case-Based Reasoning. Journal of Supply Chain Management Systems, 1(3), 40-48.

[4]. Dalal S., Agrawal A., Dahiya N., Verma J. (2020) Software Process Improvement Assessment for Cloud Application Based on Fuzzy Analytical Hierarchy Process Method. In: Gervasi O. et al. (eds) Computational Science and Its Applications – ICCSA 2020. ICCSA 2020. Lecture Notes in Computer Science, vol. 12252. Springer, Cham. https://doi.org/10.1007/978-3-030-58811-3_70

[5]. Seth B., Dalal S., Kumar R. (2019) Hybrid Homomorphic Encryption Scheme for Secure Cloud Data Storage. In: Kumar R., Wiil U. (eds) Recent Advances in Computational Intelligence. Studies in Computational Intelligence, vol 823. Springer, Cham.

[6]. S. Pradeep and Y. K. Sharma, "A Pragmatic Evaluation of Stress and Performance Testing Technologies for Web Based Applications," in 2019 Amity International Conference on Artificial Intelligence (AICAI), 2019, pp. 399–403.

[7]. Panthee, M., & Sharma, Y. K. (2019). Review of e-government implementation. International Journal of Recent Research Aspects, ISSN: 2349-7688, 6(1), 26–30.

[8]. Y. K. Sharma and M. D. Rokade, "Deep and Machine Learning Approaches for Anomaly-Based Intrusion Detection of Imbalanced Network Traffic.," IOSR Journal of Engineering, pp. 63-67, 2019.

[9]. Bijeta Seth, Surjeet Dalal, Dac-Nhuong Le, VivekJaglan, NeerajDahiya, Akshat Agrawal, Mayank Mohan Sharma, Deo Prakash, K. D. Verma, Secure Cloud Data Storage System Using Hybrid Paillier–Blowfish Algorithm, Computers, Materials & Continua, Vol.67, No.1, 2021, pp.779-798, doi:10.32604/cmc.2021.014466

[10]. Sunita Saini, Dr.Yogesh Kumar Sharma, "LI-Fi the Most Recent Innovation in Wireless Communication", International Journal of Advanced research in Computer Science and Software Engineering, Volume 6, Issue 2, February 2016.

[11]. M. M. Sakr, M. A. Tawfeeq and A. B. ElSisi, "Filter versus wrapper feature selection for network intrusion detection system," in Proc. ICICIS, Cairo, Egypt, pp. 209–214, 2019.

[12]. N. T. Pham, E. Foo, S. Suriadi, H. Jeffrey and H. F. M. Lahza, "Improving performance of intrusion detection system using ensemble methods and feature selection," in Proc. ACSW, Queensland, Australia, pp. 1–6, 2018.

[13]. N. K. Kanakarajan and K. Muniasamy, "Improving the accuracy of intrusion

© 2021 JPPW. All rights reserved

detection using gar-forest with feature selection," in Proc. FICTA, Durgapur, India, pp. 539–547, 2016.
[14]. Z. Markov and I. Russell, "An introduction to the WEKA data mining system," ACM SIGCSE Bulletin, vol. 38, no. 3, pp. 367–368, 2006.
[15]. M. Panda, A. Abraham and M. R. Patra, "Discriminative multinomial naive Bayes for network intrusion detection," in Proc. IAS, Atlanta, GA, USA, pp. 5–10, 2010.
[16]. B. Pfahringer, "Winning the KDD99 classification cup: Bagged boosting," ACM SIGKDD Explorations Newsletter, vol. 1, no. 2, pp. 65–66, 2000.
[17]. I. Levin, "KDD-99 classifier learning contest LL soft's results overview," ACM SIGKDD Explorations Newsletter, vol. 1, no. 2, pp. 67–75, 2000.
[18]. D. S. Kim and J. S. Park, "Network-based intrusion detection with support vector machines," in Proc. ICOIN, Jeju Island, Korea, pp. 747–756, 2003.
[19]. A. Chandrasekhar and K. Raghuveer, "An effective technique for intrusion detection using neuro-fuzzy and radial svm classifier," in Proc. in Computer Networks & Communications (NetCom), Toronto Canada, pp. 499–507, 2013.
[20]. S. Mukkamala, A. H. Sung and A. Abraham, "Intrusion detection using an ensemble of intelligent paradigms," Journal of Network and Computer Applications, vol. 28, no. 2, pp. 167–182, 2005.
[21]. S. Maheswaran, S. Sathesh, Gayathri, E. D. Bhaarathei and D. Kavin, "Design and development of chemical free green embedded weeder for row based crops," Journal of Green Engineering, vol. 10, no. 5, pp. 2103–2120, 2020.
[22]. S. Sathesh, V. A. Pradheep, S. Maheswaran, P. Premkumar, N. S. Gokul et al., "Computer vision based real time tracking system to identify overtaking vehicles for safety precaution using single board computer," Journal of Advanced Research in Dynamical and Control Systems, vol. 12, no. 7, pp. 1551–1561, 2020.
[23]. M. A. Ambusaidi, X. He and P. Nanda, "Unsupervised feature selection method for intrusion detection system," in Proc. Trustcom, Helsinki, Finland, vol. 1, pp. 295–301, 2015.
[24]. I. Lopez Moreno, J. Gonzalez Dominguez, D. Martinez, O. Plchot, J. Gonzalez-Rodriguez et al., "On the use of deep feedforward neural networks for automatic language identification," Computer Speech & Language, vol. 40, pp. 46–59, 2016.
[25]. K. He, X. Zhang, S. Ren and J. Sun, "Delving deep into rectifiers: surpassing human-level performance on imagenet classification," in Proc. ICCV, Santiago, Chile, pp. 1026–1034, 2015.
[26]. S. AgatonovicKustrin and R. Beresford, "Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research," Journal of Pharmaceutical and Biomedical Analysis, vol. 22, no. 5, pp. 717–727, 2000.
[27]. M. A. Salama, H. F. Eid, R. A. Ramadan, A. Darwish and A. E. Hassanien, "Hybrid intelligent intrusion detection scheme," in Proc. in Soft Computing in Industrial Applications, Ostrava, Czech Republic, pp. 293–303, 2011.
[28]. H. F. Eid, M. A. Salama, A. E. Hassanien and T. H. Kim, "Bi-layer behavioral-based feature selection approach for network intrusion classification," in Proc. FGIT, Jeju Island, Korea, pp. 195–203, 2011.
[29]. X. Zhang, J. Ran and J. Mi, "An intrusion detection system based on convolutional neural network for imbalanced network traffic," in Proc. ICCSNT, Dalian, China, pp. 456–460, 2019.
[30]. S. M. Kasongo and Y. Sun, "A deep learning method with filter based feature engineering for wireless intrusion detection system," IEEE Access, vol. 7, pp. 38597–38607, 2019.
[31]. T. Mehmod and H. B. M. Rais, "Ant colony optimization and feature selection for intrusion detection," in Proc. MALSIP, Ho Chi Min City, Vietnam, pp. 305–312, 2016.

© 2021 JPPW. All rights reserved

[32]. C. Khammassi and S. Krichen, "A GA-lR wrapper approach for feature selection in network intrusion detection," Computers & Security, vol. 70, pp. 255–277, 2017.
[33]. C. Kalimuthan and J. A. Renjit, "Review on intrusion detection using feature selection with machine learning techniques," Materials Today: Proceedings, vol. 33, pp. 3794–3802, 2020.
[34]. Y. Zhong, W. Chen, Z. Wang, Y. Chen, K. Wang et al., "HELAD: A novel network anomaly detection model based on heterogeneous ensemble learning," Computer Networks, vol. 169, pp. 107049, 2020.
[35]. Y. Xiao, C. Xing, T. Zhang and Z. Zhao, "An intrusion detection model based on feature reduction and convolutional neural networks," IEEE Access, vol. 7, pp. 42210–42219, 2019.
[36]. H. Malhotra and P. Sharma, "Intrusion detection using machine learning and feature selection," International Journal of Computer Network & Information Security, vol. 11, no. 4, pp. 3794–3802, 2019.
[37]. K. Wu, Z. Chen and W. Li, "A novel intrusion detection model for a massive network using convolutional neural networks," IEEE Access, vol. 6, pp. 50850–50859, 2018.
[38]. F. A. Khan, A. Gumaei, A. Derhab and A. Hussain, "A novel two-stage deep learning model for efficient network intrusion detection," IEEE Access, vol. 7, pp. 30373–30385, 2019.
[39]. M. Abdullah, A. Alshannaq, A. Balamash and S. Almabdy, "Enhanced intrusion detection system using feature selection method and ensemble learning algorithms," International Journal of Computer Science and Information Security, vol. 16, no. 2, pp. 48–55, 2018.
[40]. P. Gogoi, M. H. Bhuyan, D. Bhattacharyya and J. K. Kalita, "Packet and flow based network intrusion dataset," in Proc. IC3, Noida, India, pp. 322–334, 2012.
[41]. [33] S. Mukherjee and N. Sharma, "Intrusion detection using naive Bayes classifier with feature reduction," Procedia Technology, vol. 4, pp. 119–128, 2012.
[42]. E. De La Hoz, A. Ortiz, J. Ortega and E. De la Hoz, "Network anomaly classification by support vector classifiers ensemble and non-linear projection techniques," in Proc. HAIS, Salamanca, Spain, pp. 103–111, 2013.

© 2021 JPPW. All rights reserved