Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH

Open Access

Design of Intrusion Detection System based on Cyborg intelligence for security of Cloud Network Traffic of Smart Cities

Edeh Michael Onyema[1], Surjeet Dalal[2*], Carlos Andrés Tavera Romero[3], Bijeta Seth[4], Praise Young[5] and Mohd Anas Wajid[6]

Abstract

The Internet of things (IoT) is an important technology that is highly beneficial in establishing smart items, connections and cities. However, there are worries regarding security and privacy vulnerabilities in IoT in which some emerge from numerous sources, including cyberattacks, unsecured networks, data, connections or communication. This paper provides an ensemble intrusion strategy based on Cyborg Intelligence (machine learning and biological intelligence) framework to boost security of IoT enabled networks utilized for network traffic of smart cities. To do this, multiple algorithms such Random Forest, Bayesian network (BN), C5.0, CART and Artificial Neural Network were investigated to determine their usefulness in identifying threats and attacks-botnets in IoT networks based on cyborg intelligence using the KDDcup99 dataset. The results reveal that the AdaBoost ensemble learning based on Cyborg Intelligence Intrusion Detection framework facilitates dissimilar network characteristics with the capacity to swiftly identify different botnet assaults efficiently. The suggested framework has obtained good accuracy, detection rate and a decreased false positive rate in comparison to other standard methodologies. The conclusion of this study would be a valuable complement to the efforts toward protecting IoT-powered networks and the accomplishment of safer smart cities.

Keywords: Cyborg, Ensemble learning, IoT, Network Intrusion Detection System (NIDS), Cloud Computing

Introduction

Technology is fast penetrating every sector of human life, including where we live and how we work or share things with people and organization [1]. Smart cities are becoming more common as a result of a rise in global urbanization. Infrastructure and amenities are needed to balance with ecological and transportation challenges as the population grows in metropolitan areas. Smart cities are evolving as a solution to the aforementioned difficulties [2]. Developing a wide range of smart city applications has become increasingly dependent on the rapid growth and expansion of low-cost devices and other IoT-oriented infrastructure, which have been integrated with wireless communication technology [3]. Figure 1 shows variety of useful municipal services such as advanced public transit and health care, advanced construction, as well as innovative manufacturing and waste management.

The Internet of Things (IoT) is a key component of smart city technology. The Internet of Things refers to the interconnection of computer devices (smart items) via the Internet [4]. Using a variety of physical sensors and wireless networks, data is collected and analyzed in near-real time in the Internet of Things (IoT). Actuators are controlled and processed by the data collected by sensors [5]. As an outcome, the expansion of IoT infrastructure in smart cities will require the assurance of critical characteristics such security, secrecy, trust,

*Correspondence: surjeetd.computers@tmu.ac.in

[2] College of Computing Science and IT, Teerthanker Mahayeer University, Moradabad Uttar Pradesh, India
Full list of author information is available at the end of the article

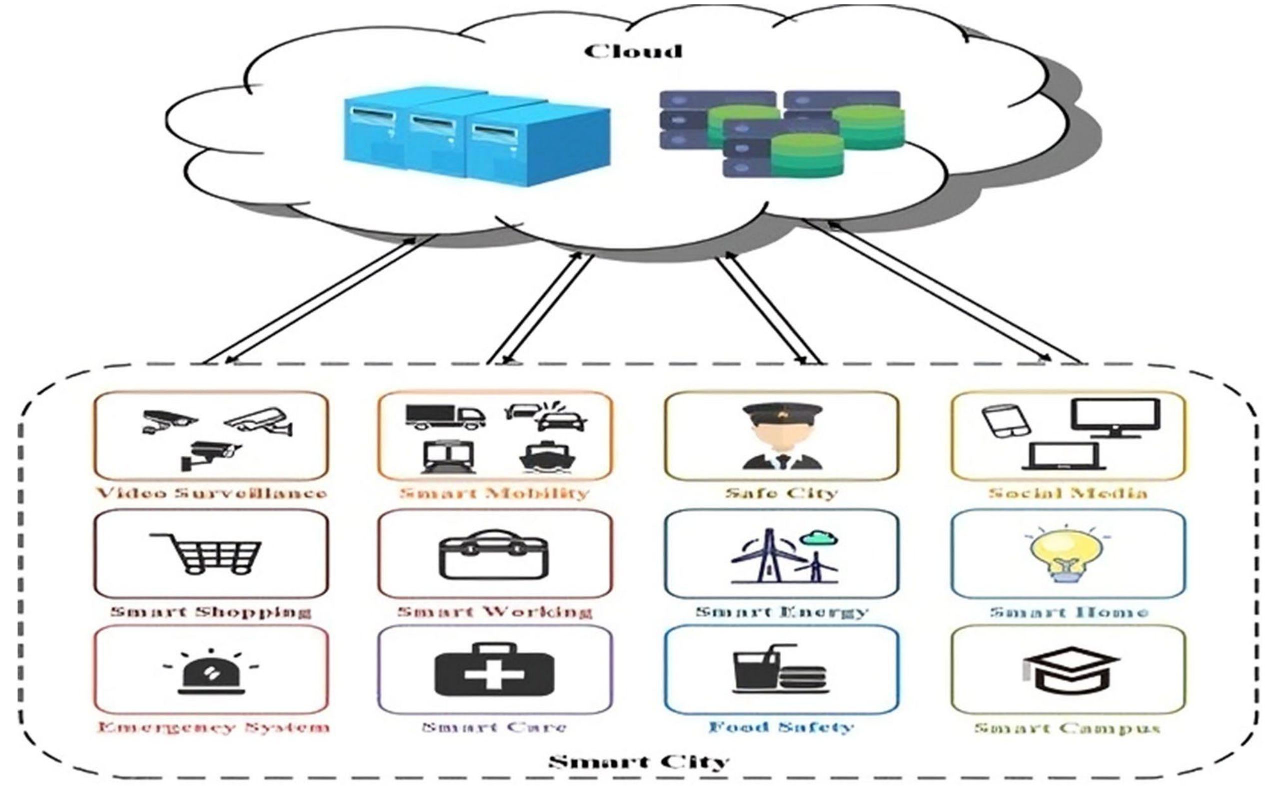


Fig. 1 Data Storage in Smart Cities

scalability, and centralization [6]. IoT systems are more scattered and diverse than traditional systems. With the unique qualities [7] of IoT like memory competency and computing capacity, network bandwidth as well as long-lasting battery life, securing, protecting, and trusting IoT has become more difficult. The interconnectedness of various IoT sensors in smart networks that target IoT devices in smart cities introduces a multitude of possible risks. Smart cities are vulnerable to both physical and cyber threats. To initiate physical assaults, hackers must be in close proximity to equipment and be able to alter/tamper with the network's hardware and software [8].

It's becoming increasingly difficult to manage the security of data particularly that of IoT devices [2]. At present, the most popular method of storing data is via the cloud. The cloud servers have powerful computational and storage resources. Clients of the cloud store their files on cloud servers, pay only for what they use, and have access to a global network of computers and servers from anywhere in the world. However, as illustrated in Fig. 1, storing data in the cloud for smart cities might be wasteful and unreliable. There are several limitations of uploading data to the cloud. Substantial time delays can occur by uploading data from IoT devices to cloud servers, data integrity may be hindered, and the loss of control over the data can occur if the local backup is deleted.

A number of challenges relating to data privacy and security exist in smart city environments, putting at risk the proper operation of smart devices and degrading the use of their services. In addition, the images and videos that can be accessed in a smart city and the information collected in smart houses may show the level of life of the customers. The ethical difficulties are the root cause of these problems. Residents' safety might be jeopardized by malicious attackers using this information. People's personal information, such as photographs, videos, and other media, might be exposed as a result. In addition, malicious attackers may be able to identify the normal behavior of people (such as their departure or arrival times) that might have an impact on the citizens' lives through data mining studies. As a result, adequate safeguards against privacy and security concerns are essential to prevent citizens from rejecting smart cities [9].

To maintain the safety and privacy of the smart cities, the acquired data should be protected from unauthorized access. A smart city collects and analyses data about its residents' well-being and the environment in which they live, and then uses that data to augment the standard of their daily lives. Consequently, safety and confidentiality issues might be considered hurdles to the smart city's degree of maturity. One of the most pressing concerns is the security of personal information and the dependability of data storage and processing.

Motivation and key challenges:

Although it isn't unexpected, there has been a lot of study done on how to protect data from cyberattacks. In support of the growth of sustainable smart cities, there are, nevertheless, a number of issues that must be addressed. Here are some of the motivational behind this research work:

1. To address the research gaps in network intrusion detection process in connected devices of Smart cities and analyze them carefully.
2. To develop a more accurate and efficient network intrusion detection mechanism for smart cities.
3. To design the above cited mechanism in context of Cyborg intelligence for getting benefits of both human and machine intelligence.
4. To implement proposed mechanism of particular dataset for judging the performance of this mechanism.
5. To assess the performance of existing machine learning algorithms in network intrusion detection.

To tackle the smart city challenges, this search paper propose a new AdaCyborg (AdaBoost ensemble learning method based on Cyborg Intelligence) Intrusion Detection for smart cities.

Overview and literature survey

This segment includes schemes used in IoT-driven smart cities and its networks and latest investigation allied to threat detection, challenges of threat detection, covering known and unknown threats, and anomaly detection system, and several machine learning algorithms.

How does threat detection differ from threat protection?

Threat detection is the repeatable process conducted in near real time, or retroactively, in order to detect and respond to adversary actions or toolsets, typically detected through conventional security controls. It is a process which is often technology, or analyst-driven, and which combines security tools, analysis, and experience.

While these two terms are sometimes used interchangeably, the reality is that they are fundamentally different. Threat protection is typically signature-based, and is designed to alert based on Indicators Of Compromise (IOCs) of malware or tools. These artefacts, typically aligning to the lower levels of Dave Bianco's Pyramid of Pain, could include things such as IP addresses, domain names, hash values, and textual strings in a file. These elements can be used for alerting, but they are "fragile" and signatures using them can break without notice if an attacker modifies their tools or changes their infrastructure, leading organizations to have a false sense of security.

Threat detection, however, aligns more to the upper levels of the Pyramid, and includes more complicated elements of malware and tools. This could include specific behaviors of malware or tools on the system or network as they attempt to establish persistence, exploit specific vulnerabilities, or communicate with their command and control (C2) servers. Detecting on these elements is more reliable, and it takes significantly more effort for adversaries to evade detection [8].

Challenges of threat detection

To be successful, threat detection should be done in real-time. However, there are many challenges that are associated with to-the-second detection. Analysts are overburdened by alerts from abundant security tools. Collecting hundreds of log types and analyzing them, even when using more sophisticated techniques including machine learning and behavioral analysis, is unsustainable for the majority of organizations. Even more so, logs lack content and context, making it difficult to parse out true threats. Though once a threat is detected, logs can help SOC teams quickly map timelines and provide analysis of the threat event:

- Best endpoint log sources for threat hunting
- Best network log sources for threat hunting

Known vs. unknown threats

To protect our environments, speed is critical. Security programs that detect threats quickly and efficiently are able to reduce the overall risk to the organization. Ideally, an organization's defense program can stop the majority of threats because the malicious acts have been spotted in the wild and their signature data in traditional threat protection platforms has been recorded—and the organization has details on how to mitigate the attack. Even still, some of these threats can slip through defensive measures, which is why SOCs should have analysts with hands on keyboards looking for threats [10]. The flip side of the coin is that the threat landscape is constantly changing and introducing new, unknown threats that have not yet been detected. To detect both known and unknown threats, defenders should use a variety of methods, including:

- **Threat Intelligence:** Effective threat intelligence is actionable, and consistently shares the traits of contextualization, evaluation, prioritization, customization and decomposition. Often, security programs focus too much on the quantity of threat intelligence, instead of the more important "quality" of the intelligence. Threat intelligence lessens the overall danger to organizations, their members and clients as comparable to rapid threat detection and response.

- **Threat Hunting:** Unlike other forms of threat detection, threat hunting is a proactive process that identifies the presence of malicious actors and their tools before an attack.

Disconnected security tools and the problem with after-the-fact

Threat detection and response is more difficult than years ago because there are number of disconnected point tools for analysts to use. However, the effectiveness of these tools is limited because each tool must be deployed, configured, and operated daily. The analysts have to work for endpoint security, network security tools, cyber threat intelligence, malware and adversaries in the network [11].

The after-the-fact detection is a problem because it generally does not happen within minutes of an attack. Only 22% detect breaches in less than one day. And when there are low detection rates, there are much longer business impacts.

Related work

Almeida et al. [10] managed a basic presentation investigation of an Internet of Things mindful Ambient Assisted Living (AAL) framework for older observing. The examination was centered around three fundamental framework parts: (I) the far reaching information catching layer, (ii) the Cloud- based unified information the executives vault, and (iii) the gamble investigation and expectation module. Every module could give different working modes, subsequently the basic examination targets characterizing which were the best arrangements as indicated by setting's necessities.

Moustafa et al. [12] proposed a gathering interruption location strategy to alleviate noxious occasions, specifically botnet assaults used in IoT organizations. In light of an investigation of their potential qualities, new factual stream highlights were derived from conventions. This was followed by the creation of an AdaBoost group learning approach that used three AI methods to analyze the influence of these aspects and differentiate malignant events. It had been determined by using the correlation and association coefficient measurements that the suggested highlights have the potential properties of either normal or noxious activity. In addition, the suggested collection approach had a greater recognition rate and a lower rate of false positives compared to the system and three other best-in-class grouping techniques.

Using the IoT/Fog/Cloud standards, miniature administrations, and DevOps foundations, Iasio et al. [11] presented the reference engineering, a model execution, and a city scale contextual investigation assessment of PROMENADE, a stage that ensured continuous improvement of strong and solid applications for ongoing checking and examination of traffic information produced by IoT gadgets in enormous intelligent urban areas. Based on on-line traffic circumstances, the model was evaluated for a scenario study on the semi-continuous detection of street network flaws using centrality measures derived from unconnected real datasets available for Lyon, France.

Using a sophisticated nonparametric Bayesian model, Makkar [13] proposed to create a discovery model for both known and obscure interruptions (or irregularity location). Our system's design might be expanded to meet the needs of IoT innovation as well as impressively smart city online applications. They used a Bayesian-based MCMC induction for infinitely limited summed up Gaussian blend models to familiarize ourselves with the examples of the exercises (typical and atypical). In spite of exemplary grouping techniques, our methodology did not have to indicate the quantity of bunches, thinks about the vulnerability through the presentation of earlier information for the boundaries of the model, and allows to tackle issues connected with over-and under-fitting. To get better grouping execution, include loads, model's boundaries, and the quantity of bunches were assessed at the same time and consequently. The created approach was assessed utilizing famous informational indexes. The acquired outcomes showed the proficiency of our methodology in recognizing different assaults.

Insightful electronic devices (IEDs) with built-in communicated interruption location frameworks had been proposed by Hong et al. [7]. For example, tested values and conventional items located in substations, the suggested IEDs could screen for and detect anomalies and weird ways of acting in the host arrangement of IED and IEC 61,850-based messages. As a group, the suggested IEDs aimed to pinpoint the start of digital attacks by coordinating with adjacent IEDs. Using the item inserted framework, the suggested interruption discovery framework was tested using IEDs' power framework insurance features. Overcurrent and distance protections on the installed board could be reliably and effectively mitigated by the provided moderation approaches.

The Trustworthy Privacy-Preserving Secured Framework (TP2SF) was developed by Kumar et al. [9] for smart urban communities. An interruption location module, a reliability module, and two-level security are all included in this system. Address-based blockchain renown will be implemented in the reliability module. Two-level security uses a blockchain-based upgraded Proof of Work (ePoW) approach and Principal Component investigation (PCA) to transform information into a reduced form to prevent derivation and damaging attacks. An improved inclination tree support framework (XGBoost) was provided in the interruption locating module. Final results of the Fog-Cloud design have led us to provide a blockchain-IPFS coordinated basis for the Fog-Cloud, notably CloudBlock and Fogblock, to express the suggested TP2SF system in the shining city. Both in non-blockchain and blockchain settings, the findings demonstrated the superiority of TP2SF structure over other cutting-edge approaches.

It had been developed by Bhayo et al. [14] to identify DDoS attacks based on the counter advantages of various organizational boundaries. C-DAD was a flexible and adaptable system that had been thoroughly tested across a wide range of organizational boundaries. The SDN-enhanced findings were clearly seen in the computation. Aside from that, the proposed design distinguished the assault in a shorter amount of time and with less strain on the computer's processor and memory.

Guo et al. [15] proposed an AI based strategy that can distinguish explicit weak IoT gadget models associated behind a homegrown NAT, in this way recognizing home organizations that represented a gamble to the telcos framework and administration accessibility. To assess our strategy, they gathered a huge amount of organization traffic information from different business IoT gadgets in our lab and thought about a few grouping calculations. We discovered that our stream-based method is powerful and can cope with situations where previous NAT-detection strategies fail, such as scrambled, non-TCP or non-DNS traffic, which can't be properly addressed by existing NAT-detection strategies. We've made our original marked benchmark dataset available for others to use in their research.

Nie et al. [16] utilized head part investigation (PCA) for include decrease and group based classifiers are utilized to foresee interruption assaults on the organizations. KDDCup-'99' dataset has been utilized and execution is assessed as far as exactness, accuracy, review and F-score. Nonetheless, in the shrewd medical care climate, IoMT gadgets face high weakness. Network safety was a fundamental part of a shrewd city that could accomplish a protected climate for savvy medical services. Accordingly, the Intrusion Detection System (IDS) was utilized as a security layer of correspondence towards online protection for the most recent gadgets and organizations frameworks.

Elsaeidy et al. [17] fostered a profound learning-based model for replay assault recognition in savvy urban communities. This model's performance was evaluated by comparing it to a real-world Smart City dataset, where attacks were re-enacted and replayed. According to their findings, the suggested model had the ability to accurately identify both normal and threatening ways of acting. In addition, the results showed that the suggested model outperformed both traditional order and deep learning methods. A real smart city informative indexed with imitated replay assaults was also included as an added commitment for future study.

Lee et al. [18] used Gated Recurrent Unit cells to uncover correlations between time series information, and used Gaussian Mixture priors in inactive space to depict multimodal information. The inability to fully capture information designs had been caused by previous efforts expecting simple conveyances for Gaussian Mixture priors. With the help of the Bayesian Inference Criterion (BIC), they proposed a model selection instrument throughout the preparation cycle to observe the model that can accurately measure conveyance in Gaussian Mixture (GM) space. On four datasets, they conducted extensive replications and discovered that our suggested plot outperforms cutting-edge peculiarity identification algorithms, improving F1scores by up to 47.88 percent. An information-driven IDS was being planned by Ahmad et al. [5] by researching the RSU's connection load methods of acting in the IoV against various attacks that caused traffic streams to vacillate. Convolutional Neural Network (CNN) was being used to separate the components of connection stacks and pinpoint the point of interruption centered on RSUs. As a result of the back-propagationcomputation, the suggested engineering was made up of a standard CNN and a simple blunder term. In the meantime, the proposed CNN-based profound designwas provided a hypothetical investigation by the probabilistic portrayal. IoT devices may be protected by using ML to identify spam, according to Seth et al. [18]. Spam Detection in IoT using Machine Learning system was presented to achieve this purpose. Five machine learning models were evaluated in this system using a wide range of measurements and data highlighting sets from a wide range of information sources. The enhanced information highlights are taken into consideration by each model when registering a spam score. The reliability of an IoT device may be measured by this score. The REFIT Smart Home dataset is used to provide the go-ahead to a new strategic plan. When compared to alternative options, the proposed scheme is more viable, according to the results obtained.

A Privacy Preserving and Secure Framework (PPSF) for IoT-driven brilliant cities is presented by Kumar et al. [3]. The PPSF relied on a two-level protection conspiracy and an interruption identification plan as the foundations of its design. Two-level protection begins with a blockchain module and Principal Component Analysis (PCA) employed to transform unrefined IoT data into a more manageable form. Two IoT network datasets, namely ToN-IoT and BoT-IoT, were analyzed using a Gradient-Boosting Anomaly Detector (GBAD) in the interruption location plot to prepare and evaluate the suggested two-level protection conspiracy. To deliver the suggested PPSF structure, we also advocated an IPFS-integrated Fog- Cloud blockchain architecture. Exploratory outcomes showed the predominance of the PPSF structure over a few late methodologies in blockchain and non-blockchain frameworks.

Wan et al. [19] fostered an IoT traffic estimation structure on programmable and smart edge switches to consequently gather approaching, active, and inward organization traffic of IoT gadgets in edge organizations, and to fabricate multi-faceted social profiles which portray who, when, what, and why on the personal conduct standards of IoT gadgets in light of persistently gathered traffic information.

Shahraki et al. [20] surveyed the information stream handling instruments and structures that could be utilized to handle such information on the web or on-the-fly alongside their upsides and downsides, and their integrality with de truth information handling systems. To investigate the presentation of OL procedures, we lead an experimental assessment on the exhibition of various gathering and tree-based calculations for network traffic characterization. At last, the open issues and the future headings in investigating traffic information streams were introduced. This specialized review presented important bits of knowledge and standpoint for the organization research local area while managing the necessities and motivations behind web-based information streams examination and learning in the systems administration space.

Ahmed et al. [8] proposed a heap adjusting calculation to plan sensor information, vehicles and server farms performing assignments. They likewise offered a bundle level interruption recognition model. The outcomes demonstrated the way that the created model could further develop the choice limit by utilizing a pooling procedure and an entropy vulnerability measure. Table 1 summarizes the recent work done in this problem domain.

Materials and methods

Dataset

The research work has taken KDDCUP'99 data set for experimental work. It is most commonly used data sets available for network-based inconsistency detection systems. This dataset has 42 columns of various data types.

Overview of Methods involved and algorithms chosen

Below is an outline of involved algorithms and steps.

Data Pre-processing

Data collected from varied sources collected in a raw format need to be converted before analysis. Data pre-processing or data cleaning refers to the process used to transform raw data into a clean data set. It is an essential step in machine learning because it is said that "Better data beats fancier algorithms".

Missing values

Different cleaning methods are required for different types of data. Missing data needs to be handled carefully in machine learning as they can be relevant. The missing data can be handled in two ways but may result in sub-optimal information:

Table 1 Related Work Summery

No	Paper	Key Features
1	[10]	Ambient Assisted Living (AAL) framework developed
2	[12]	AdaBoost group learning approach, three AI methods were used
3	[11]	Traffic analysis through PROMENADE was done
4	[10]	Irregularity location was discussed, employed Bayesian-based MCMC induction
5	[7]	Insightful electronic devices (IEDs) was discussed
6	[9]	Described interruption location module, a reliability module, and two-level security were included, addressed Proof of Work
7	[14]	identified DDoS attacks
8	[19]	IoT traffic estimation structure on programmable and smart edge switches were mentioned
9	[3]	Privacy Preserving and Secure Framework (PPSF) was discussed, PCA
10	[20]	Discussed information stream handling instruments and structures, network traffic
11	[8]	Employed heap adjusting calculation

- Dropping observations with missing values: It is sub-optimal as dropping observations may lead to drop information which may be informative.
- Imputing the missing values from past observations: It is sub-optimal and may lead to loss of information as the value was originally missing but we filled it in.

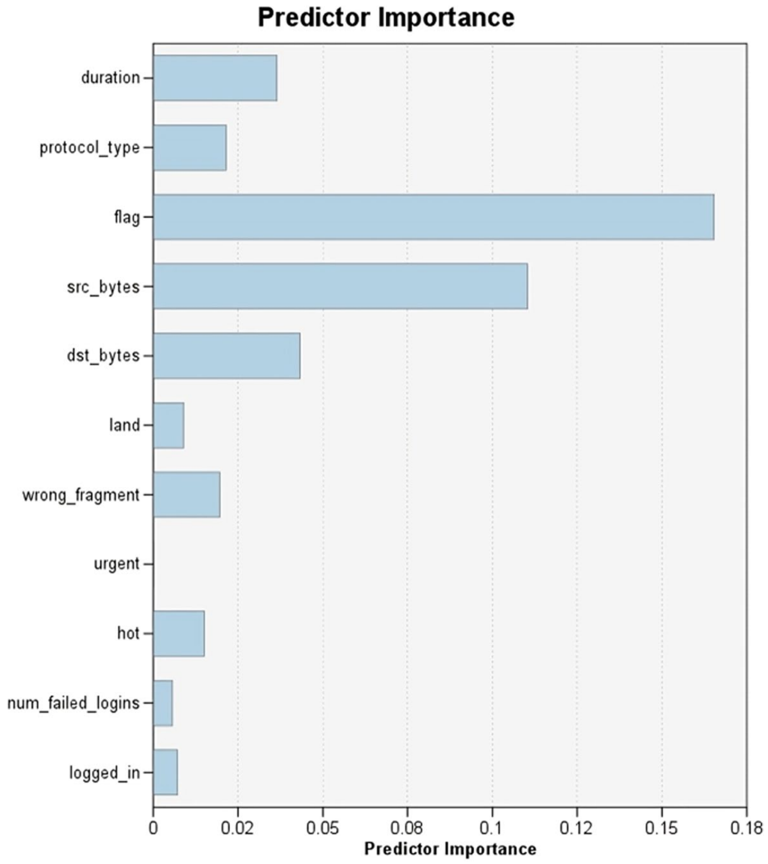
Outliers data Outliers are the values which look different from the other values in the data. They may occur because of a faulty sensor or an error in data entry and any natural discrepancy like wrong input of salaries. Outliers can create problems during model fitting or inflate the error metrics.

Random forest for classification

Random forest, given by Tin Kam Ho in 1995, is a supervised machine learning algorithm employed for classification and regression problems. It builds decision trees on various samples and the majority vote for performing classification and average vote in case of regression. For instance, checking an email is spam or not [21–25]. It uses random subspace method. An extension of algorithm was developed by Leo Breiman and Adda Cutter in 2006 using "Bagging" idea. It is depicted in Fig. 2.

RF model has following advantages as given below:

1. Bagging algorithm and Ensemble learning approach are the main advantages. On the data subset, it produces as many trees as possible and assembles their results. Overfitting and variance may be reduced using this method. As a result, it enhances accuracy.
2. Classification and regression issues are both addressed by this technique.
3. It is able to handle both categorical and continuous variables.
4. Missing values can be handled automatically.
5. Because it relies on a rule-based method rather than distance calculations, no standardization or normalization, i.e. feature scaling, is necessary.
6. Non-linear parameters may be handled well using this tool.



The top 10 inputs are shown.

Fig. 2 Random Tree model details

7. It is incredibly stable and has a low influence on the surrounding environment.

Table 2 shows the decision rules used for attack.
Disadvantages

1. The algorithm increases the overall complexity as compared to decision trees due to high computational power and resources.
2. Longer training period is required as it produces many trees versus a single tree in case of decision tree.

C5.0

A C5.0 model divides the sample into subsets based on the most informative field. Ross Quinlan was the brains behind it. The procedure repeats itself until the subsamples can no longer be divided further, at which point the process ends [26–29]. Figure 3 shows the specifics. Models of two distinct types may be generated using C5.0:

- Decision tree: This tree can only make one prediction based on a single piece of data.
- Rule set: Because they are based on decision trees, they have simpler mathematical models.

Table 2 Top Decision Rules for 'attack' in RF model

Decision Rule	Most Frequent Category	Rule Accuracy	Ensemble Accuracy	Interestingness Index
(src_bytes < = 1032.0) &(dst_bytes > 4100.0) & (dst_host_srv_count > 168.0)	normal	0.998	0.998	0.994
(dst_host_same_srv_rate > 0.02) &(serror_rate > 0.0) & (dst_bytes < = 4100.0) &(same_srv_rate > 0.09) & (num_file_creations < = 0.0)	apache2	0.346	0.977	0.124
(src_bytes > 0.0) &(hot < = 0.0) &(flag = {RSTR,S2,S3,SF}) &(duration > 0.0) &(protocol_type = {icmp,tcp})	apache2	0.210	0.479	0.073
(src_bytes > 0.0) &(dst_bytes < = 0.0) & (land < = 0.0) &(hot < = 0.0) &(protocol_type = {icmp,tcp})	apache2	0.192	0.688	0.067
(duration > 0.0) &(num_root < = 0.0) & (src_bytes > 0.0) &(flag = {RSTR,S0,S2,S3,SF,SH}) & (land < = 0.0)	apache2	0.173	0.572	0.059

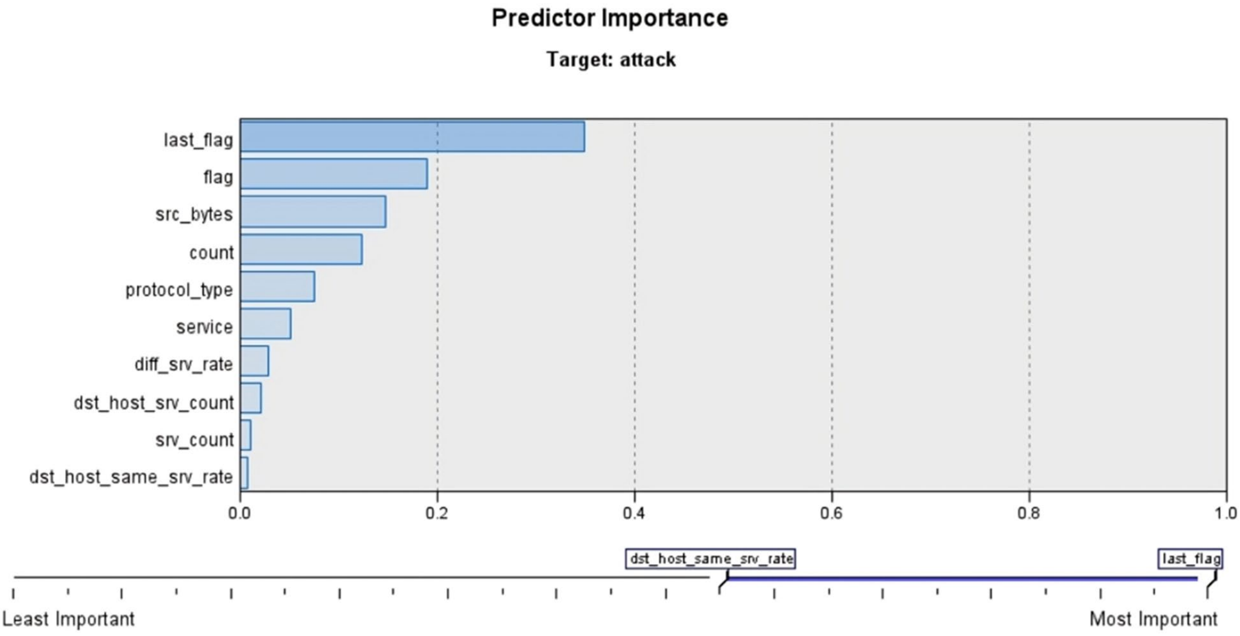


Fig. 3 C5.0 model details

Figure 3 highlights the predicator importance in C5.0 model. It has following advantages as given below:

- C5.0 is quite robust in case of missing data and large numbers of input files. They don't require long training times to estimate.
- C5.0 is easy to understand as the rules derived from the model have straightforward interpretation.

Neural network

A mathematical or computer model based on biological neural networks is known as an artificial neural network (ANN). It utilizes a connectionist approach to computation and is made up of a network of artificial neurons. Nodes and weighted edges are the building blocks of a neural network [30–32]. Recurrent and feed forward neural networks can be distinguished in general. Figure 4 depicts its specifications.

It has following advantages as given below:

1. Ability to train machines: In artificial neural networks, comparable events are used to learn and make judgments.
2. Parallel processing ability: An artificial neural network's numerical strength enables it to simultaneously accomplish a variety of tasks.

Figure 5 demonstrates the neural network design. It has some limitations also. Neural Networks employ "black box" nature which result in greater computational burden, proneness to overfitting, and the empirical nature of model development.

CART

Gini's impurity index serves as the splitting criteria in the Classification and Regression Trees algorithm, which is a classification procedure for creating a decision tree. Leo

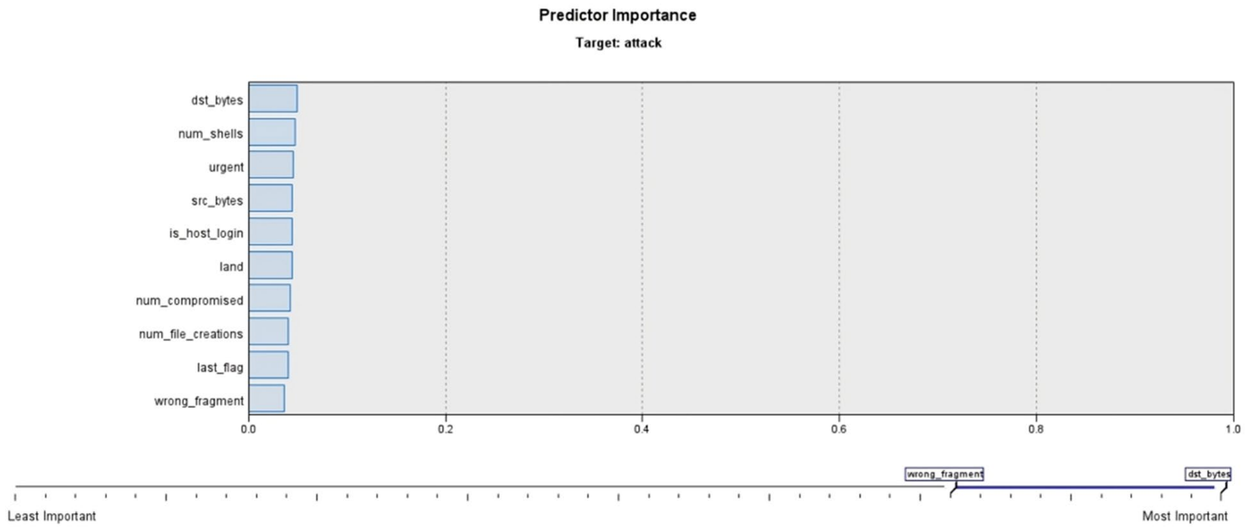


Fig. 4 Neural Network details

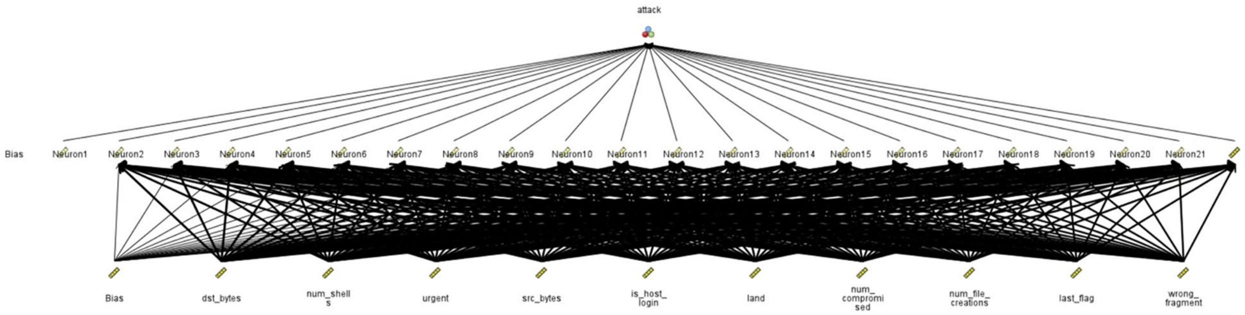


Fig. 5 Neural Network design

Braiman used the term "decision tree" to describe the method. Machine learning makes use of a type of predictive model that describes how the values of target variables may be predicted from the values of other variables. Forks in the CART output are separated into predictor variables, and each node in the end is given a predicted value. It is depicted in Fig. 6.

Advantages of CART

- However, unlike CHAID, CART always creates binary splits.
- Because CHAID constructs segments/deciles of continuous variables that are often arbitrary in nature and can sometimes mask the underlying patterns in the continuous variables, CART can test many more cut points than CHAID when dealing with continuous data.
- In comparison to CHAID, CART has a significantly smaller tree structure.
- Unlike CHAID, which naturally generates broad trees, CART develops tall and narrow trees.

Disadvantages of CART are that it is computationally expensive and slow in nature.

CHAID

CHAID represents Chi_Square Automatic Interaction Detector. It is the oldest decision tree algorithm in history. CART and CHAID both use tree-like structures to represent data, but they differ in their efforts to limit the development of the trees in the models. GordanKass invented it in 1980 and is used to uncover the correlations between variables in a statistical model CART is more commonly used for forecasting than CHAID for descriptive analysis. Figure 7 provides information about CHAID.

Advantages:

CHAID algorithm effectively generates multi-way frequency tables. It has been used in marketing research.

Bayesian network

A supervised learning method based on the Bayes theorem, the nave Bayes algorithm is used to solve classification issues. Judea Pearl created the term "Bayesian network" in 1985. Text categorization using a large, multi-dimensional training set makes extensive use of this technique. Nodes in Bayesian networks represent variables in the Bayesian senseTheyare directed acyclic graphs (DAGs). Figure 8 explains this. Figure 9 depicts Bayesian Networks' conditioned probabilities.

Advantages:

1. It provides a way of combining prior information with data.
2. It provides interpretable answers.
3. It gives a convenient setting for a large number of models.

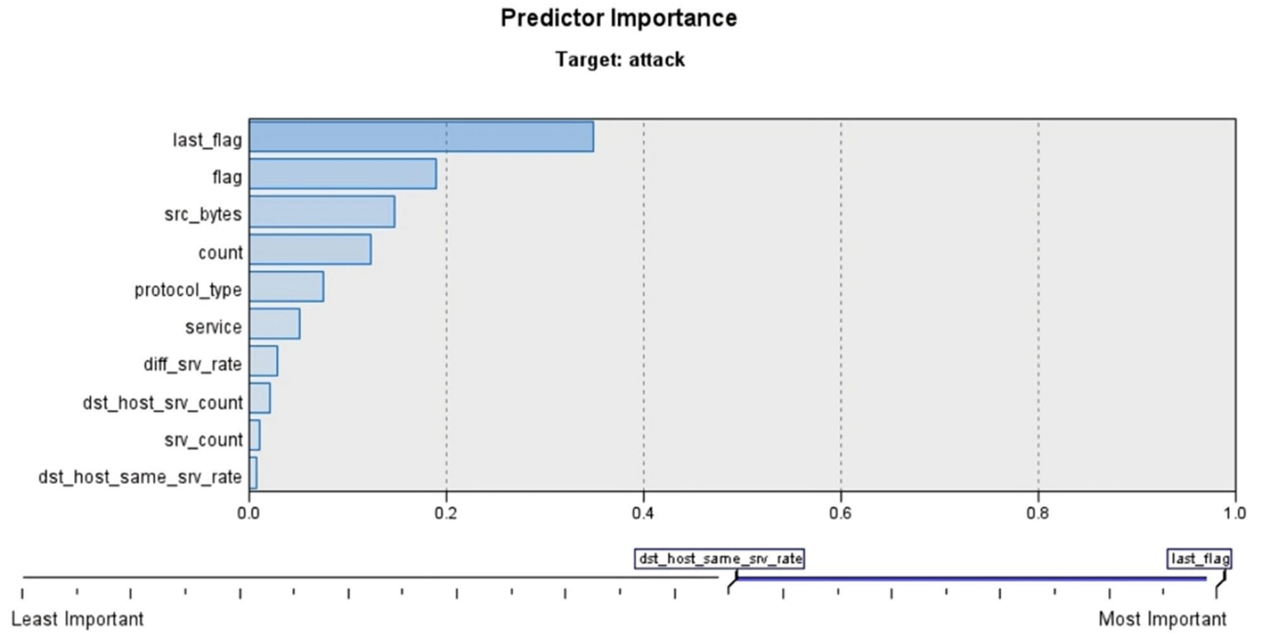


Fig. 6 CART model details

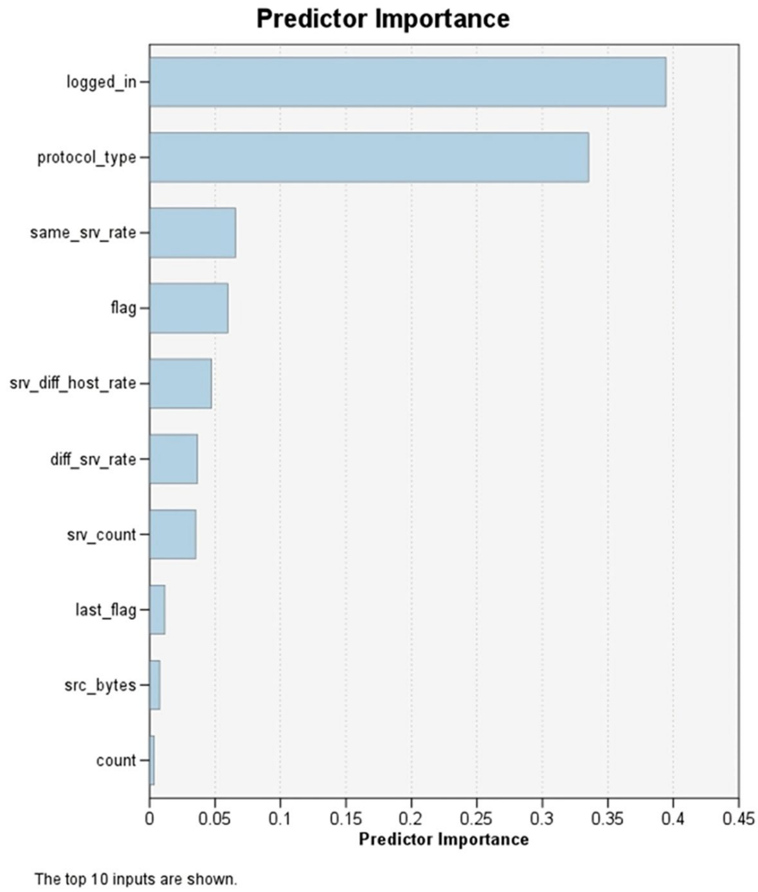


Fig. 7 CHAID details

Figure 9 highlights the Condition Probabilities for Bayesian Network. It has following disadvantages as given below:

- There is no method to choose a prior.
- It often comes with a high computational cost.

Proposed AdaCyborg intrusion detection technique

Cyborg intelligence is a new research paradigm that strives to blend the best of both biological and machine intelligence. For example, we can then compare the cyborg model's behavior to that driven by each of the two mechanisms on their own to see whether there are any similarities or differences in the model's behavior. The biological and computer portions are tightly intertwined.

Collaborations represents a long term integrated process which occurs when organizations interact formally and informally.Gray described collaboration as "a process through which parties who seedifferent aspects of a problem can constructively explore their differences and searchfor solutions that go beyond their own limited vision of what is possible". According to brain computer integration requirements, a *motivation* could berepresented as a 3-tuples {N, G, I}, where N means needs, G is goal, I means themotivation intensity [13].

There are three kinds of needs in the Cyborg system:

- Perception needs: Acquire environment information through vision, audition, touch, taste, and smell.
- Adaptation needs: Adapt environment condition and optimize impaction of action taken.
- Cooperation needs: Promise to reward a cooperation action between brain and machine.

A *motivation* is activated by motivational rules which structure has following format:

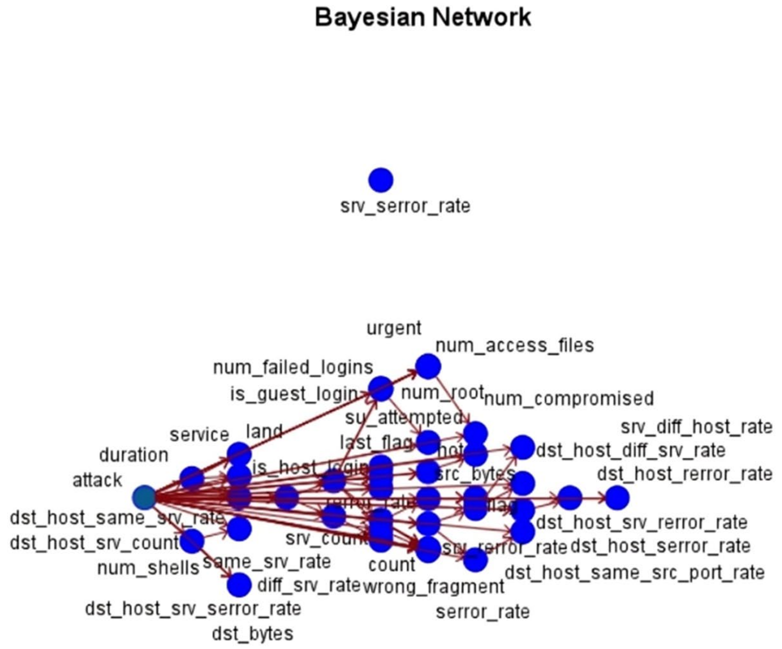
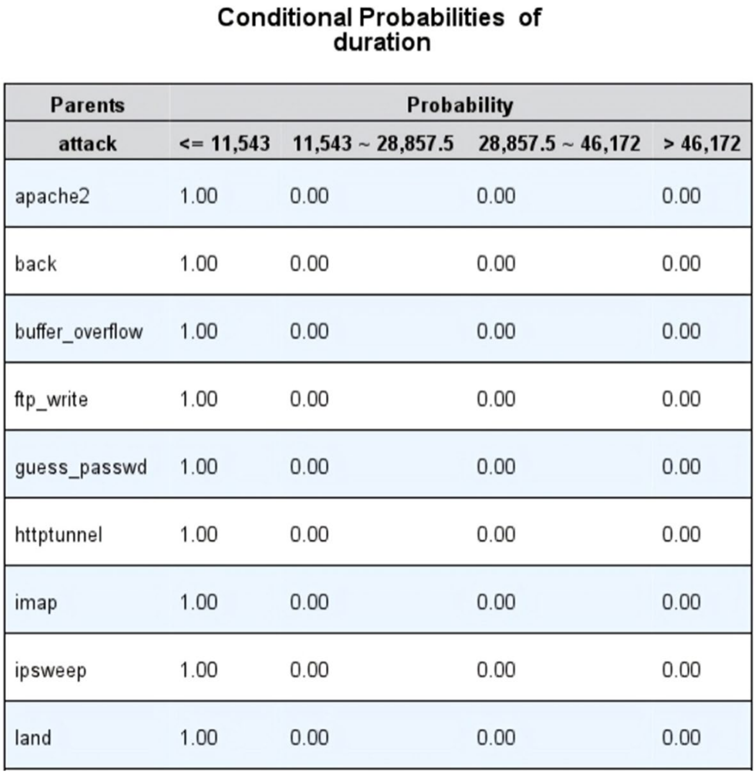


Fig. 8 Bayesian Network Details



Conditional Probabilities of duration

Parents	Probability			
attack	<= 11,543	11,543 ~ 28,857.5	28,857.5 ~ 46,172	> 46,172
apache2	1.00	0.00	0.00	0.00
back	1.00	0.00	0.00	0.00
buffer_overflow	1.00	0.00	0.00	0.00
ftp_write	1.00	0.00	0.00	0.00
guess_passwd	1.00	0.00	0.00	0.00
httptunnel	1.00	0.00	0.00	0.00
imap	1.00	0.00	0.00	0.00
ipsweep	1.00	0.00	0.00	0.00
land	1.00	0.00	0.00	0.00

Fig. 9 Condition Probabilities for Bayesian Network

$$R = \left(P, D, \mathrm{Strength}(P|D)\right)$$

where, P indicates the conditions of rule activation; D is a set of actions for the motivation; Strength(P|D) is a value within interval [0,1]. Based on the Adaboost approach, a network intrusion detection system is proposed in this research.

The proposed algorithm has following steps as given below:

Step 1—First of all network connection vector will be assigned some weights. Initially, all the weights will be equal. It helps to define Perception needs of proposed system.
Step 2—To select the important features among these network traffic attributes, the correlation coefficient has been applied in this step. This step belongs to Feature selection and defines Adaptation needs.
Step 3—Next the decision stump has been created for each feature with respect to target classes. It describe the Cooperation needs.
Step 4 – Next stage, the classification is done based on motivational rules which is based on conditions of rule activation and Strength.
Step 5 – Final decision will be decided on outcome of step 4.
Step 6 – Exit

An outline of the proposed work is shown in a flowchart below in Fig. 10.

The first step is to choose features from the KDD95 dataset, which contains a pool of features. For this purpose, we perform a correlation analysis. We eliminate those features that are substantially connected with each other from the results of the inquiry. Classifying network traffic as benign or dangerous is also covered in this study.

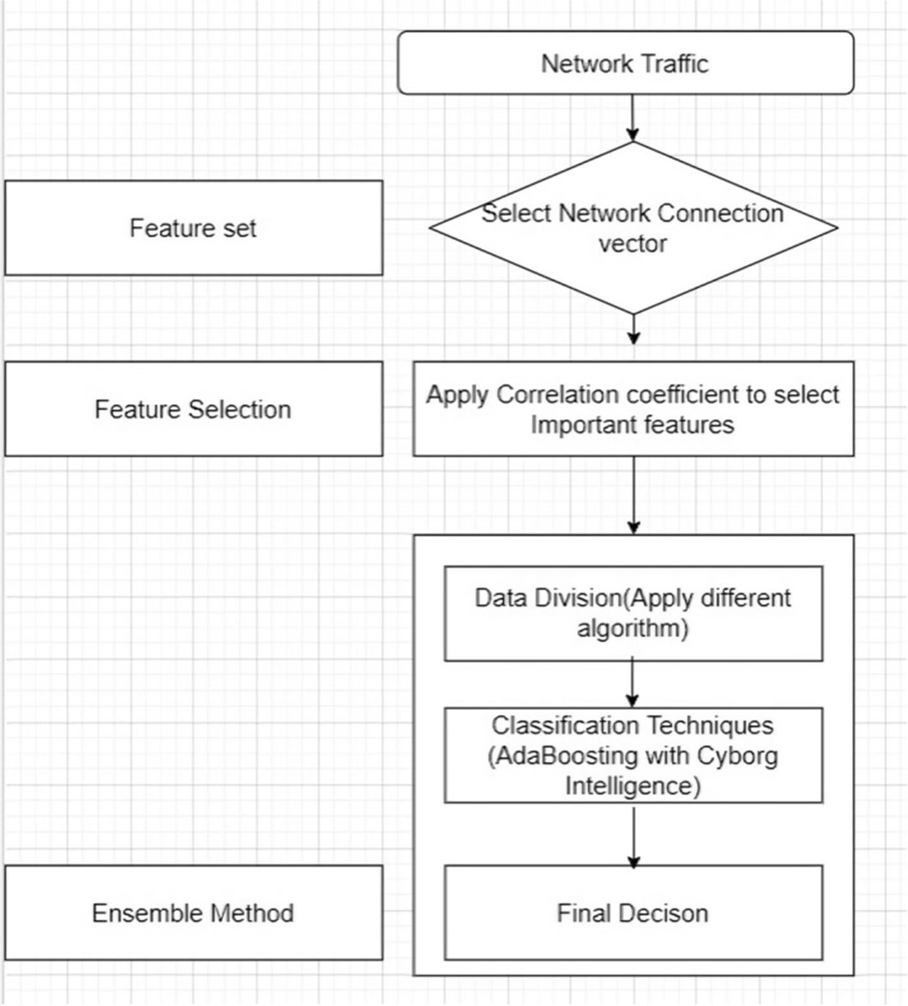

Network Traffic
Feature set
Select Network Connection vector
Feature Selection
Apply Correlation coefficient to select Important features
Data Division(Apply different algorithm)
Classification Techniques (AdaBoosting with Cyborg Intelligence)
Ensemble Method
Final Decison

Fig. 10 Flowchart of AdaBoost algorithm

Experimental results and discussion

The AdaCyborg (Cyborg Intelligence based ensemble model) is implemented in Python by utilizing package AdaBoost Scikit-Learn API for building decision tree base learners. The proposed system was applied to KDDCUP'99 data set. A large number of experiments were implemented using varied parameter settings detailed in the next sections.

Machine learning algorithms in network intrusion detection

In the first phase, we implemented various machine learning algorithms discussed above to detect the network attacks. This section provides the accuracy level maintained in them individually.

Figure 11 demonstrated the accuracy level in Random Forest model. It has secured 84.29% accuracy on given dataset.

Figure 12 demonstrated the accuracy level in C 5.0 model. It has secured 83.94% accuracy on given dataset.

Figure 13 demonstrated the accuracy level in CART model. It has secured 99.43% accuracy on given dataset.

Figure 14 demonstrated the accuracy level in CHAID model. It has secured 76.91% accuracy on given dataset.

Figure 15 demonstrated the accuracy level in Bayesian Networkmodel. It has secured 95.84% accuracy on given dataset.

Figure 16 demonstrated the accuracy level in Bayesian Network model. It has secured 98.24% accuracy on given dataset.

Proposed AdaCyborg model in network intrusion detection

Figure 17 demonstrated the accuracy level in the proposed AdaCyborg model. It has secured 99.43% accuracy on given dataset.

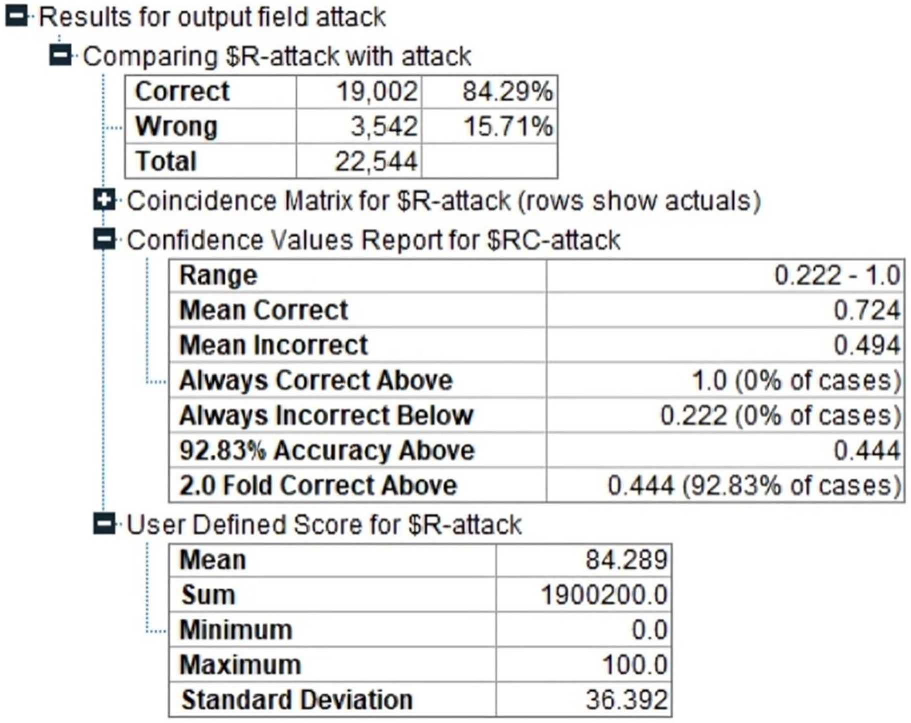
As demonstrated in the simulations, the suggested technique has an excellent detection rate and low false positive rate. Table 3 highlights the comparison of accuracy of proposed model with existing works.

Additionally, AdaCyborg's detection time is short, demonstrating its efficiency. The Fig. 18 demonstrates the results graphically for better understanding.

Security both in the cloud and IoT would remain an important factor in achieving smart city and in maximizing the potentials of IoT –powered solutions [38–41].

Complexity

Complexity is calculated in terms of space and time. Time complexity is the computational complexity that



Results for output field attack

Comparing $R-attack with attack

Correct	19,002	84.29%
Wrong	3,542	15.71%
Total	22,544	

Coincidence Matrix for $R-attack (rows show actuals)

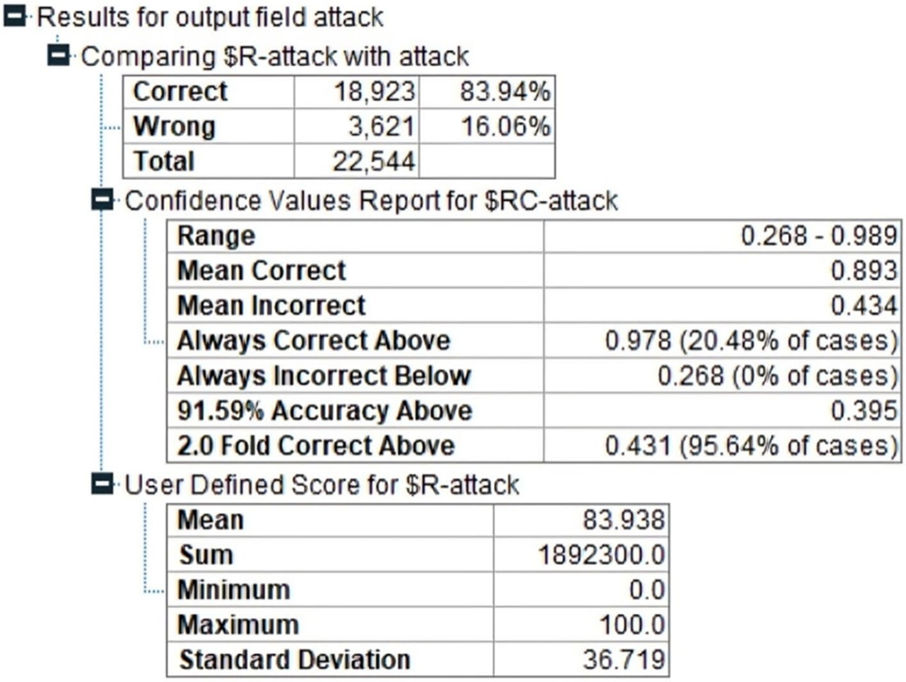
Confidence Values Report for $RC-attack

Range	0.222 - 1.0
Mean Correct	0.724
Mean Incorrect	0.494
Always Correct Above	1.0 (0% of cases)
Always Incorrect Below	0.222 (0% of cases)
92.83% Accuracy Above	0.444
2.0 Fold Correct Above	0.444 (92.83% of cases)

User Defined Score for $R-attack

Mean	84.289
Sum	1900200.0
Minimum	0.0
Maximum	100.0
Standard Deviation	36.392

Fig. 11 Accuracy level in Random Forest model



Results for output field attack

Comparing $R-attack with attack

Correct	18,923	83.94%
Wrong	3,621	16.06%
Total	22,544	

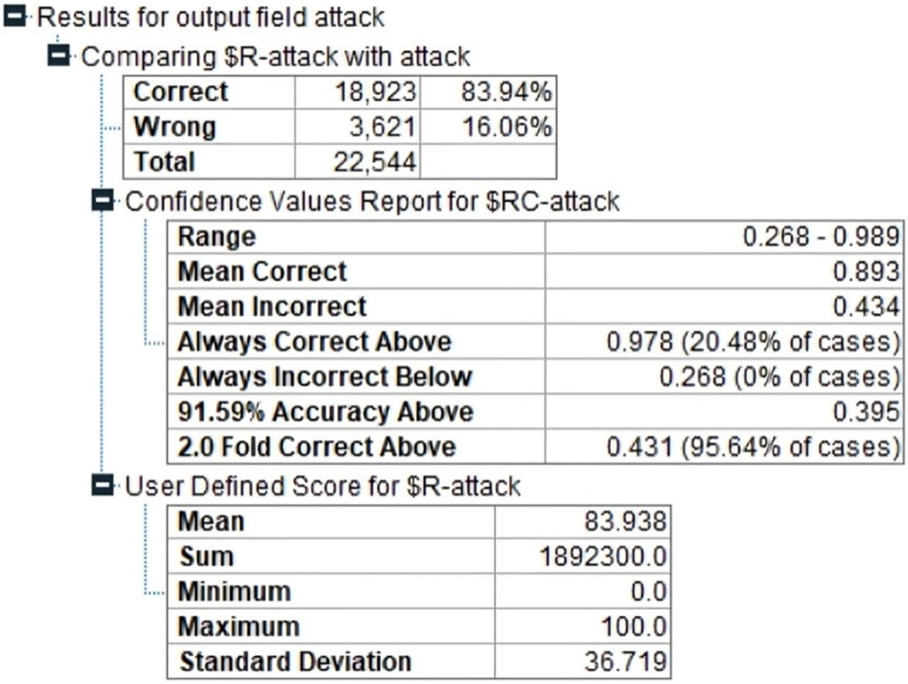
Confidence Values Report for $RC-attack

Range	0.268 - 0.989
Mean Correct	0.893
Mean Incorrect	0.434
Always Correct Above	0.978 (20.48% of cases)
Always Incorrect Below	0.268 (0% of cases)
91.59% Accuracy Above	0.395
2.0 Fold Correct Above	0.431 (95.64% of cases)

User Defined Score for $R-attack

Mean	83.938
Sum	1892300.0
Minimum	0.0
Maximum	100.0
Standard Deviation	36.719

Fig. 12 Accuracy level in C 5.0 model



Results for output field attack

Comparing $R-attack with attack

Correct	18,923	83.94%
Wrong	3,621	16.06%
Total	22,544	

Confidence Values Report for $RC-attack

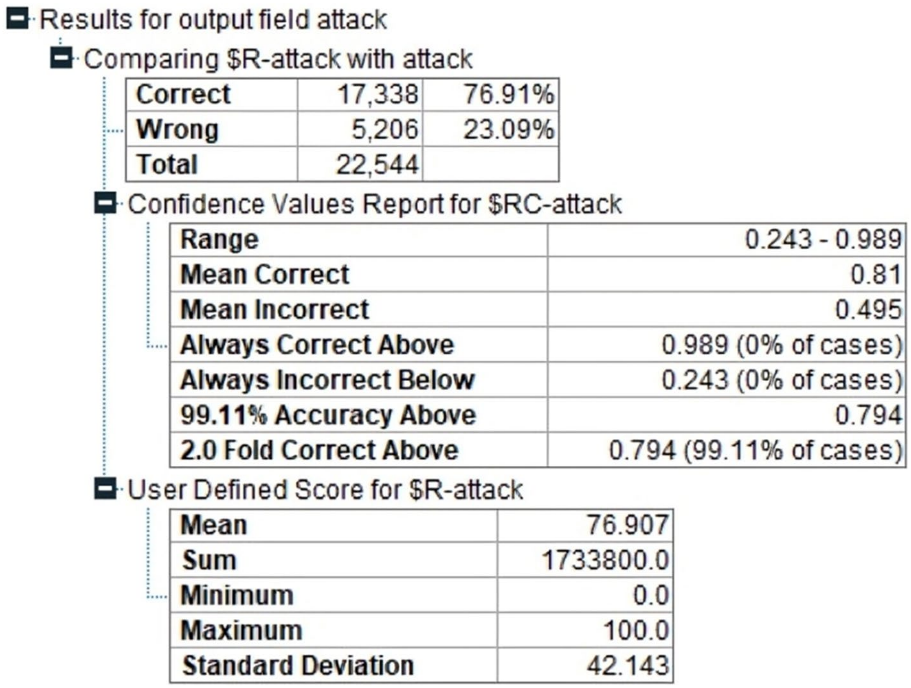
Range	0.268 - 0.989
Mean Correct	0.893
Mean Incorrect	0.434
Always Correct Above	0.978 (20.48% of cases)
Always Incorrect Below	0.268 (0% of cases)
91.59% Accuracy Above	0.395
2.0 Fold Correct Above	0.431 (95.64% of cases)

User Defined Score for $R-attack

Mean	83.938
Sum	1892300.0
Minimum	0.0
Maximum	100.0
Standard Deviation	36.719

Fig. 13 Accuracy level in CART model

describes the amount of time it takes to run an algorithm. It is estimated by counting the number of elementary operations performed by the algorithm, supposing that each elementary operation takes a fixed amount of time to perform. Thus, the amount of time taken and the number of elementary operations performed by the algorithm



Results for output field attack

Comparing $R-attack with attack

Correct	17,338	76.91%
Wrong	5,206	23.09%
Total	22,544	

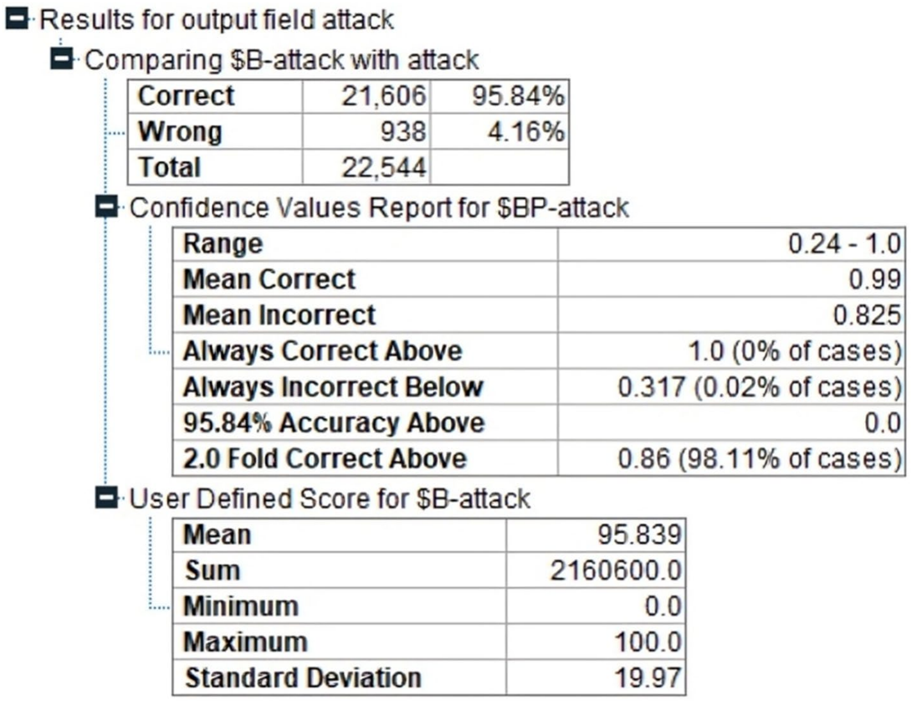
Confidence Values Report for $RC-attack

Range	0.243 - 0.989
Mean Correct	0.81
Mean Incorrect	0.495
Always Correct Above	0.989 (0% of cases)
Always Incorrect Below	0.243 (0% of cases)
99.11% Accuracy Above	0.794
2.0 Fold Correct Above	0.794 (99.11% of cases)

User Defined Score for $R-attack

Mean	76.907
Sum	1733800.0
Minimum	0.0
Maximum	100.0
Standard Deviation	42.143

Fig. 14 Accuracy level in CHAID model



Results for output field attack

Comparing $B-attack with attack

Correct	21,606	95.84%
Wrong	938	4.16%
Total	22,544	

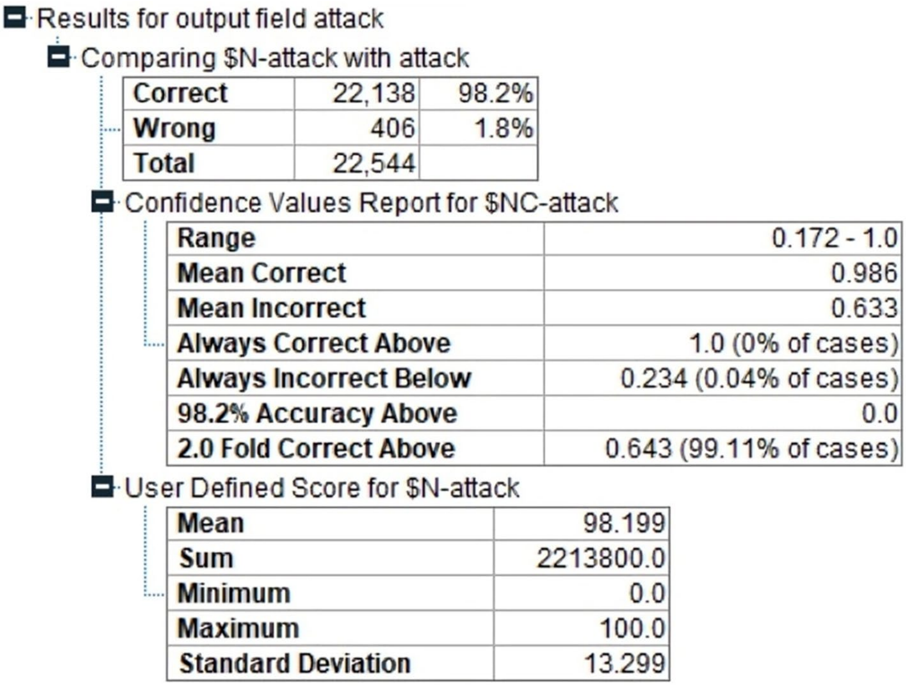
Confidence Values Report for $BP-attack

Range	0.24 - 1.0
Mean Correct	0.99
Mean Incorrect	0.825
Always Correct Above	1.0 (0% of cases)
Always Incorrect Below	0.317 (0.02% of cases)
95.84% Accuracy Above	0.0
2.0 Fold Correct Above	0.86 (98.11% of cases)

User Defined Score for $B-attack

Mean	95.839
Sum	2160600.0
Minimum	0.0
Maximum	100.0
Standard Deviation	19.97

Fig. 15 accuracy level in Bayesian Network model



Results for output field attack

Comparing $N-attack with attack

Correct	22,138	98.2%
Wrong	406	1.8%
Total	22,544	

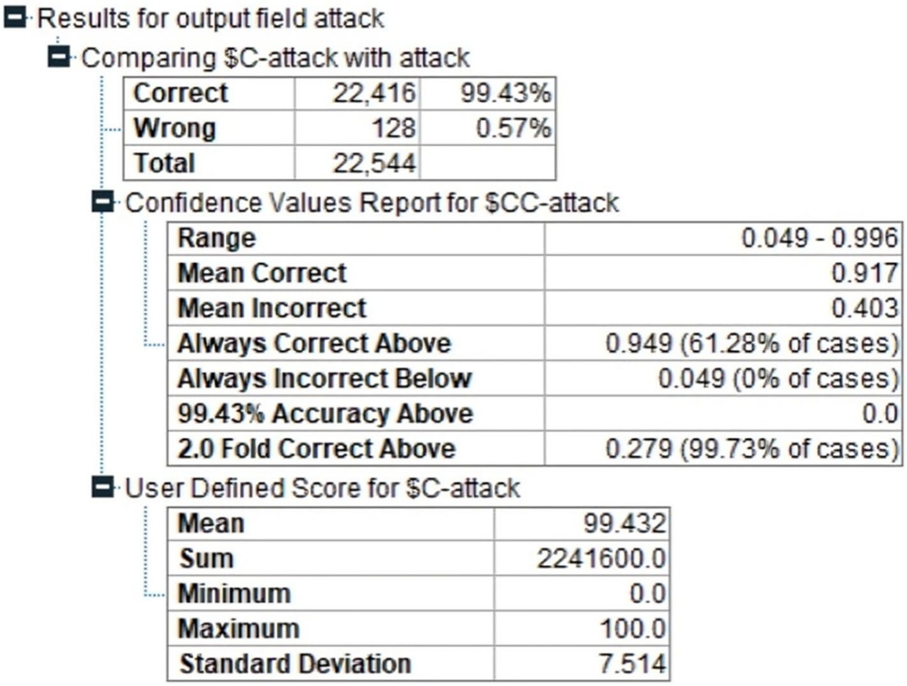
Confidence Values Report for $NC-attack

Range	0.172 - 1.0
Mean Correct	0.986
Mean Incorrect	0.633
Always Correct Above	1.0 (0% of cases)
Always Incorrect Below	0.234 (0.04% of cases)
98.2% Accuracy Above	0.0
2.0 Fold Correct Above	0.643 (99.11% of cases)

User Defined Score for $N-attack

Mean	98.199
Sum	2213800.0
Minimum	0.0
Maximum	100.0
Standard Deviation	13.299

Fig. 16 accuracy level in Neural Network model



Results for output field attack

Comparing $C-attack with attack

Correct	22,416	99.43%
Wrong	128	0.57%
Total	22,544	

Confidence Values Report for $CC-attack

Range	0.049 - 0.996
Mean Correct	0.917
Mean Incorrect	0.403
Always Correct Above	0.949 (61.28% of cases)
Always Incorrect Below	0.049 (0% of cases)
99.43% Accuracy Above	0.0
2.0 Fold Correct Above	0.279 (99.73% of cases)

User Defined Score for $C-attack

Mean	99.432
Sum	2241600.0
Minimum	0.0
Maximum	100.0
Standard Deviation	7.514

Fig. 17 accuracy level in Proposed AdaCybrog model

Table 3 Results comparison with existing works

S. No	Paper	Method	Accuracy
1	[33]	AdaBoost	92.3
2	[34]	SVM	97.56
3	[35]	Neural Network	99.33
4	[36]	Random forest	99.0
5	[37]	Deep Learning	97.2
6	Proposed AdaCyborg model	Cyborg Intelligence	99.43

is taken to differ by at most a constant factor. Space complexity is a measure of the amount of working storage an algorithm needs. Computational Complexity of Adaboost is estimated using following:

- Train time complexity: O(n*p*n_trees)
- Test time complexity: O(p*n_trees)

Here n denotes the data points in the dataset, p the number of features and n_trees suggest the number of estimators we are using in our model.

While training the model we are going through n variables for p features to find out the decision stump, and then calculate their loss function and to update the weights in this way we are taking O(n*p) operations. Now the same is repeated for n_trees number of estimators so it takes O(n*p*n_trees) operations.

The study inferred that security issues in the cloud have persisted but the proposed technique offers a good opportunity to enhance the security measures in Cloud Network Traffic of Smart Cities. The presented model in this study can help halt malicious goals in Cloud Network traffic. The system could be very effective in protecting users' privacy and the integrity of resources within the network. The study proved that vulnerability of cloud networks can be reduced using instruction detection system particularly as it relates to cyborg intelligence.

Conclusions and future work

Intrusion detection is an important and challenging problem that has a major impact on quality and reliability of smart city services. To this extent, botnet attack has been one of the most common threats on smart city infrastructure. The biological brain and machine learning algorithms must operate together in order for this integration to be successful and co-adaptive. Experiment results show that accuracy measurements may be used to detect both normal and malicious behavior. We have chosen AdaBoost algorithm as it involves very short (one-level) decision trees. The weak learners are added sequentially to the ensemble. Each subsequent model attempts to correct the prediction made earlier by the model. This is achieved by weighing the training dataset to put more focus on training examples on which prior models made prediction errors. A budding notion, Cyborg intelligence, utilizes the best traits of both machine and biological intelligences have shown to have a significant influence on Network Intrusion Detection. According to the results, our proposed AdaBoost ensemble learning based on Cyborg Intelligence Intrusion Detection framework outperforms traditional classification and deep learning models. For future work, we want to develop a prototype of the algorithm in a real-world context, potentially inside a tight network of connected computers. This will allow us to more effectively evaluate its real-world utility.

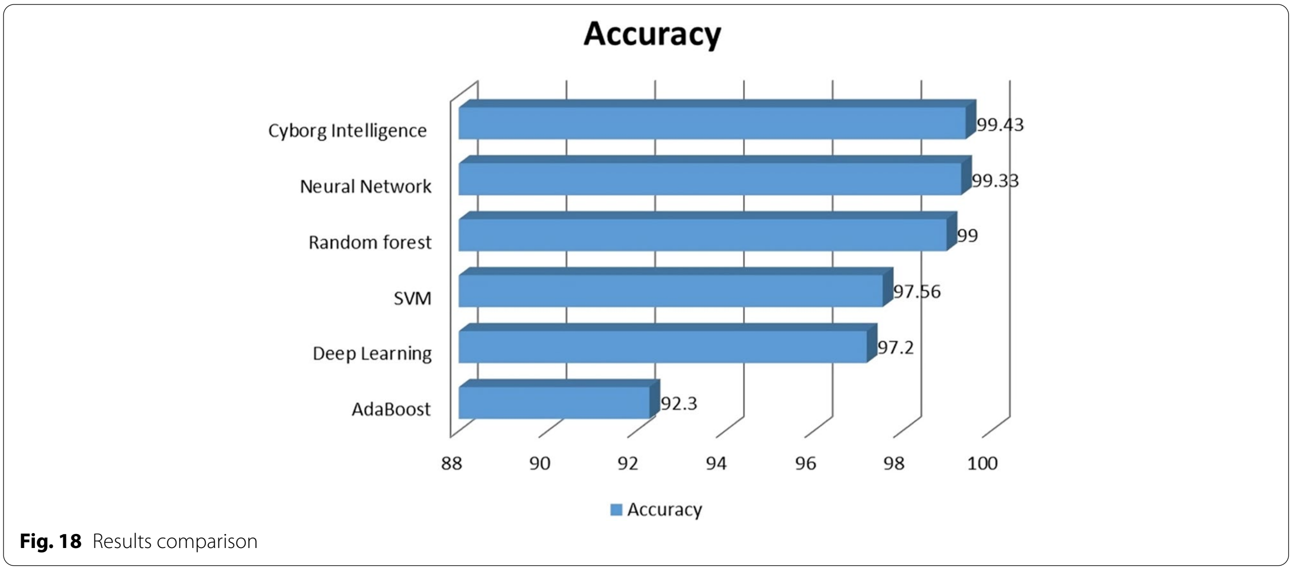


Fig. 18 Results comparison

Acknowledgements
Not applicable.

Authors' contributions
Edeh Michael Onyema: Results and analysis. Surjeet Dalal: Design, methods, submission and corresponding author. Carlos Andrés Tavera Romero: Conceptualization, Discussion and review of the final draft. Bijeta Seth: Introduction and background. Praise Young: Conclusion and background. Mohd Anas Wajid: Conclusion and review of the first draft. The author(s) read and approved the final manuscript.

Funding
This research was funded by Dirección General de Investigaciones of Universidad Santiago de Cali under call No. 01–2022.

Availability of data and materials
The supporting data may be provided by corresponding author on request.

Declarations

Ethics approval and consent to participate
There is no ethical approval required and authors express their consent to participate for paper which is a health care based on Blockchain and Privacy Computing.

Consent for publication
Authors provide the consent for publication which is a health care based on Blockchain and Privacy Computing.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Department of Mathematics and Computer Science, Coal City University, Enugu, Nigeria. [2]College of Computing Science and IT, Teerthanker Mahayeer University, Moradabad Uttar Pradesh, India. [3]COMBA R&D Laboratory, Faculty of Engineering, Universidad Santiago de Cali, 760000 Cali, Colombia. [4]Department of Computer Science and Engineering, B. M. Institute of Engineering & Technology, Sonipat, Haryana, India. [5]Department of Linguistic Data Sciences, University of Eastern Finland, 80101 Joesuu, Finland. [6]Department of Computer Science, Aligarh Muslim University, Aligarh 202002, India.

Received: 21 May 2022 Accepted: 29 July 2022
Published online: 13 August 2022

References

1. Onyema EM, Edeh CD, Gregory US, Edmond VU, Charles AC, Richard-Nnabu NE (2021) Cybersecurity awareness among undergraduate students in Enugu Nigeria. Int J Inform Sec Privacy Digital Forensic 5(1):34–42
2. Rockson KA, Michael A, Onyema EM (2020) Implementing morpheme-based compression security mechanism in distributed systems. Int J Innov Rese Dev 9(2):157–162
3. Kumar P et al (2021) PPSF: a privacy-preserving and secure framework using blockchain-based machine- learning for IoT-Driven smart Cities. IEEE Trans Netw Sci Eng 8(3):2326–2341. https://doi.org/10.1109/TNSE.2021.3089435
4. Alhakami W, Alharbi A, Bourouis S, Alroobaea R, Bouguila N (2019) Network anomaly intrusion detection using a nonparametric Bayesian approach and feature selection". IEEE Access 7:52181–52190. https://doi.org/10.1109/ACCESS.2019.2912115
5. Ahmad I, Haq QEU, Imran M, Alassafi MO, Alghamdi RA (2022) An efficient network intrusion detection and classification system. Mathematics 10(3):1–15. https://doi.org/10.3390/math10030530
6. Saba T. (2020) Intrusion Detection in Smart City Hospitals using Ensemble Classifiers, Proc. - Int. Conf. Dev. eSystems Eng. DeSE 2020-December:418–422. https://doi.org/10.1109/DeSE51703.2020.9450247
7. Hong J, Liu CC (2019) Intelligent electronic devices with collaborative intrusion detection systems". IEEE Trans Smart Grid 10(1):271–281. https://doi.org/10.1109/TSG.2017.2737826
8. Ahmed U, Lin JCW, Srivastava G, Yun U, Singh AK (Accepted/In press). Deep Active Learning Intrusion Detection and Load Balancing in Software-Defined Vehicular Networks. IEEE Transactions on Intelligent Transportation Systems. https://doi.org/10.1109/TITS.2022.3166864.
9. Kumar P, Gupta GP, Tripathi R (2021) TP2SF: a trustworthy privacy-preserving secured framework for sustainable smart cities by leveraging blockchain and machine learning. J Syst Archit 115:101954. https://doi.org/10.1016/j.sysarc.2020.101954
10. Almeida A, Mulero R, Rametta P, Urošević V, Andrić M, Patrono L (2019) A critical analysis of an IoT—aware AAL system for elderly monitoring". Futur Gener Comput Syst 97:598–619. https://doi.org/10.1016/j.future.2019.03.019
11. A De Iasio, A Futno, L Goglia, E Zimeo (2019) A Microservices Platform for Monitoring and Analysis of IoT Traffic Data in Smart Cities". Proc - 2019 IEEE Int Conf Big Data Big Data. 5223–5232. https://doi.org/10.1109/BigData47090.2019.9006025.
12. Moustafa N, Turnbull B, Choo KKR (2019) An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things". IEEE Internet Things J 6(3):4815–4830. https://doi.org/10.1109/JIOT.2018.2871719
13. Makkar A, Garg S, Kumar N, Hossain MS, Ghoneim A, Alrashoud M (2021) An efficient spam detection technique for IoT devices using machine learning". IEEE Trans Ind Informatics 17(2):903–912. https://doi.org/10.1109/TII.2020.2968927
14. Bhayo J, Hameed S, Shah SA (2020) An efficient counter-based DDoS attack detection framework leveraging software defined IoT (SD-IoT). IEEE Access 8:2020. https://doi.org/10.1109/ACCESS.2020.3043082
15. Guo Y, Ji T, Wang Q, Yu L, Min G, Li P (2020) Unsupervised anomaly detection in IoT systems for smart cities. IEEE trans Netw Sci Eng 7(4):2231–2242. https://doi.org/10.1109/TNSE.2020.3027543
16. Nie L, Ning Z, Wang X, Hu X, Cheng J, Li Y (2020) Data-driven intrusion detection for intelligent internet of vehicles: a deep convolutional neural network-based method". IEEE Trans Netw Sci Eng 7(4):2219–2230. https://doi.org/10.1109/TNSE.2020.2990984
17. Elsaeidy AA, Jagannath N, Sanchis AG, Jamalipour A, Munasinghe KS (2020) Replay attack detection in smart cities using deep learning. IEEE Access 8:137825–137837. https://doi.org/10.1109/ACCESS.2020.3012411
18. Le D-N, Seth B, Dalal S (2018) A hybrid approach of secret sharing with fragmentation and encryption in cloud environment for securing outsourced medical database: a revolutionary approach",. J Cyber Secur Mobility 7(4):379–408
19. Wan Y, Xu K, Wang F, Xue G (2021) Characterizing and mining traffic patterns of IoT devices in edge networks". IEEE Trans Netw Sci Eng 8(1):89–101. https://doi.org/10.1109/TNSE.2020.3026961
20. Shahraki A, Abbasi M, Taherkordi A, Jurcut AD (2021) A comparative study on online machine learning techniques for network traffic streams analysis. Comput Networks 2021(207):108836. https://doi.org/10.1016/j.comnet.2022.108836
21. Meidan Y, Sachidananda V, Peng H, Sagron R, Elovici Y, Shabtai A (2020) A novel approach for detecting vulnerable IoT devices connected behind a home NAT. Comput Secur 97:101968. https://doi.org/10.1016/j.cose.2020.101968
22. Yuan Y, Huo L, Yuan Y, Wang Z (2019) Semi- supervised tri-Adaboost algorithm for network intrusion detection". Int J Distrib Sens Networks 15(6):2019. https://doi.org/10.1177/1550147719846052
23. Shi et al Z. (2017) Brain-Machine Collaboration for Cyborg Intelligence Brain-Machine Collaboration for Cyborg Intelligence, HAL Id : hal-01615002
24. 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. https://doi.org/10.1007/978-3-030-12500- 4_5
25. Ramu SP, Boopalan P, Pham QV, Maddikunta PKR, Huynh-The T, Alazab M, Nguyen TT, Gadekallu TR (2022) Federated learning enabled digital twins for smart cities: Concepts, recent advances, and future directions. Sustain Cities Soc 79:103663. ISSN 2210-6707

26. Aidan Fuller, Zhong Fan, Charles Day And Chris Barlow, " Digital Twin: Enabling Technologies, Challenges and Open Research", IEEE Acess, 2020.
27. Papyshev G, Yarime M (2021) Exploring city digital twins as policy tools: a task-based approach to generating synthetic data on urban mobility. Data & Policy 3:E16. https://doi.org/10.1017/dap.2021.17
28. Babu R, Mohammed & Ravi, Vinayakumar & Kp, Soman,"RNNSecureNet: Recurrent neural networks for Cybersecurity use-cases", https://doi.org/10.13140/RG.2.2.21876.81283, 2018
29. Marek Pawlicki, Rafał Kozik, Michał Choraś (2022) "A survey on neural networks for (cyber-) security and (cyber-) security of neural networks". Neurocomputing. 500:075–1087, ISSN 0925–2312
30. Bhattacharya S, S SRK, Maddikunta PKR, Kaluri R, Singh S, Gadekallu TR, Alazab M, Tariq U (2020) A Novel PCA-Firefly Based XGBoost Classification Model for Intrusion Detection in Networks Using GPU. Electronics 9(2):219. https://doi.org/10.3390/electronics9020219
31. Chaitanya Gupta, Ishita Johri, Kathiravan Srinivasan, Yuh-Chung Hu, Saeed Mian Qaisar, Kuo-Yi Huang,"A Systematic Review on Machine Learning and Deep Learning Models for Electronic Information Security in Mobile Networks", Sensors (Basel). 2022; 22(5):2017. https://doi.org/10.3390/s22052017
32. Mohan PV, Dixit S, Gyaneshwar A, Chadha U, Srinivasan K, Seo JT (2022) Leveraging computational intelligence techniques for defensive deception: a review, recent advances, open problems and future directions. Sensors (Basel) 22(6):2194
33. Usman Ahmed, Jerry Chun-Wei Lin, Gautam Srivastava (2022) A resource allocation deep active learning based on load balancer for network intrusion detection in SDN sensors. Comput Communications, 184:56–63.
34. Gaber T, El-Ghamry A, Hassanien AE (2022) Injection attack detection using machine learning for smart IoT applications. Physical Communication 52:101685 ISSN 1874-4907, 1-14
35. Verma A, Ranga V (2018) Statistical analysis of CIDDS-001 dataset for network intrusion detection systems using distance-based machine learning. Procedia Comput Sci 125:709–716 ISSN 1877-0509
36. Abdallah EE, Eleisah W, Otoom AF (2022) Intrusion Detection Systems using supervised machine learning techniques: a survey. Procedia Comput Sci 201:205–212 ISSN 1877-0509
37. Ashiku L, Dagli C (2021) Network intrusion detection system using deep learning. Procedia Computer Sci 185:239–247 ISSN 1877-0509
38. Onyema EM, Nwafor CE, Ugwugbo AN, Rockson KA, Ogbonnaya UN (2020) Cloud security challenges: implication on education. Int J Comput Sci Mobile Comput 9(2):56–73
39. Celestine I, P, Suresh, (2019) An efficient and unique TF/IDF algorithmic model-based data analysis for handling applications with big data streaming. Electronics 8:28
40. Iwendi C, Uddin M, Ansere JA, Nkurunziza P, Anajemba JH, Bashir AK (2018) On Detection of Sybil Attack in Large-Scale VANETs Using Spider-Monkey Technique, in IEEE Access 6:47258–47267. https://doi.org/10.1109/ACCESS.2018.2864111.
41. Iwendi C, Maddikunta PKR, Gadekallu TR, Lakshmanna K, Bashir AK, Piran MJ (2021) A Metaheuristic Optimization approach for energy efficiency in the IoT networks. Software: Pract Exp 51(12):2558–2571

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.