International Journal of Applied Engineering Research ISSN 0973-4562 Volume 18, Number 4 (2023) pp. 313-321
© Research India Publications. http://www.ripublication.com

Hybrid Machine Learning approach for Cloud Security using Genetic Algorithm

[1]Munish Saran, [2]Upendra Nath Tripathi
[1]Research Scholar, [2]Associate Professor
Department of Computer Science, DDU Gorakhpur University, Gorakhpur-273001,
Uttar Pradesh, India.
*Email: munishsaran@gmail.com**

Abstract

Cloud computing is a computing paradigm for providing the on-demand, scalable, pay-as-per-use and secure services to the end users over the internet. This computing paradigm is surrounded by various types of attacks that causes the threat to the consumer data. In this respect some research gap has been highlighted by which has to be taken care of in order to construct an IDS such as the detection of the known as well as the unknown attacks with low accuracy and high false alarm rate are the biggest challenge of the IDS based systems, selecting of most appropriate machine learning algorithms for IDS from the wide list of variants available and lastly as most IDS methodologies proposed by the researcher are based on very complex models that require a lot of time in processing and computing resources and may result in extra overhead for the processing unit and ultimately affects the performance of IDS.This paper proposes a hybrid model for the purpose of detecting the various cloud attacks from the incoming network traffic. The hybrid model is composed of genetic algorithm and decision tree algorithm (GA and CART). Genetic algorithm is used to select the most appropriate features for the machine learning model which in turns optimizes the input for the decision tree classification algorithm. The proposed approach utilizes decision tree algorithm for classifying the incoming network traffic as malicious or normal, thus allowing only the non-malicious request to reach the cloud servers. The hybrid model is trained on "UNSW-NB15" authentic dataset. The simulation results proved that the proposed hybrid model achieves 99.89% classification accuracy, 99.17% precision, 98.98% true positive rate at the same time reduces the false positive rate to 0.94%.

Keywords - Cloud Computing, Decision Tree, Genetic Algorithm, Support Vector Machine, Hyperparameter, UNSW-NB15.

1 INTRODUCTION

Data stored over the cloud is the most important asset of the end user, which has to be secured with the most secure mechanism by the cloud service providers. Today even the big cloud service providers such as Amazon, IBM, Google, Microsoft etc. which have huge infrastructure, suffers from several cloud attacks. Various technologies are involved in order to have a strong defence mechanism against the cloud attacks. Machine learning finds a key role towards the process of achieving the secure cloud ecosystem. ML has high advantage in various fields including cloud computing and finds utility for solving various problems. One of such area is to automate the process of identifying the various cloud attacks or threats. Implementing an IDS (intrusion detection system) by training the ML model with authentic cloud-attack datasets proves out to be very useful for the providers as these trained models sanitize the incoming traffic to the servers as normal or malicious thus protecting from the cloud attack. The dataset used for the purpose of training the machine learning model is the most important ingredient in order to achieve the high learning accuracy and allow the model to have deep insights of the data. Dataset contains several types of data for the purpose of analysis. But not every type of data collected in the dataset are directly relevant for the outcome of result. Some data are noisy, some are irrelevant while some may be redundant and these lead to the improper training of the machine learning model thereby reducing the decision making ability of the ML model.

Feature selection is the most important step of data science before the training of the model on the given dataset starts. It filters out the most relevant subset of feature that are highly required for the purpose of accurate decision making by the chosen machine learning model. This is true because if a model trained with most optimal features subset, it will learn more in depth and can make more effective predictions. The existing feature selection technique for supervised learning can be divided into two categories namely filter and wrapper. Filter selection is the fastest feature selection method and selects the most relevant feature set according to the calculated feature score irrespective of the ML model. Statistical techniques are used for evaluating the relationship between dependent and independent variables in the filter techniques. In contrast with the Wrapper techniques which are computationally slow and selects the high score feature subset by performing the feature selection on the various so-formed feature subsets using the classification algorithm. Optimal feature selection is not an easy task to be accomplished. As the total number of subset of features possible in a given dataset are 2^n, where n is the total number of available features. Metaheuristics algorithms such as genetic algorithm are able to select the most optimal feature set among the 2^n features and overcoming the problem of overfitting and underfitting thus leading to high performance ML model. So for this sake, our proposed framework utilizes the power of genetic algorithm in order to provide the relevant features as

input to the classification algorithm. The optimal feature subset so obtained is tested against several supervised learning based classification algorithm

as decision tree, logistic regression, support vector machine, k-nearest neighbor and naïve bayes. The rest of the paper is organized as follows: section 2 describes the theoretical background, section 3 describes the literature review, section 4 deals with the proposed work, section 5 describes the performance matrix while the environment Setup, dataset and results is described in section 6 with section 7 as the conclusion for the work proposed.

2 Theoretical Background

This section briefly describes about the theoretical background of the major technologies involved in composing this hybrid approach, i.e. Machine Learning, Cloud Computing and Genetic Algorithm.

2.1 Cloud computing- Cloud computing is a computing paradigm which enables on-demand provisioning of various computing services to the end users. The computing services provided by the service providers varies from infrastructure services provided by Amazon Web Services, Google Compute Engine, Microsoft Azure etc to platform services provided by Force.com , Windows Azure, Heroku etc and software services provided by Cisco WebEx, Salesforce, Dropbox etc [1]. NIST defines the essential features of cloud computing as resource pooling, on-demand self-service, rapid elasticity, broad network access and measured service.

- **Resource Pooling**- The underlying resources are shared between multiple clients as and when needed.
- **On-Demand Self-Service**- The cloud enabled services are accessed by the consumers on their demand as these services are automated in nature and can be invoked by the consumers.
- **Rapid Elasticity**- Depending on the consumers demand for the resources at any given point of time, cloud resources are scaled up or scaled down to meet the requirements.
- **Measured Service**- The consumers are billable for only the services they use at any time by the service providers.
- **Broad Network Access**- The access to cloud services are possible to broad range of devices.

It is due to these characteristics this computing model proves out to be beneficial to businesses, organizations, individuals, students, researchers. As this is a third party provisioning, it is surrounded by various threats and risks which can be broadly classified under five categories as:- Identity security, Information security, Infrastructure security, Network security and Software security [2]. The overall security of the cloud infrastructure is guaranteed if all of the above mentioned security categories are secured by the service providers. Some of the attack that takes place at the SaaS cloud model includes denial of service, distributed denial of service, authentication attacks, SQL injection, cross-site scripting etc. Attacks at the PaaS layer includes phishing, man-in-the-middle, cloud malware injection, password reset, etc. While some attacks that occur at the IaaS cloud layer includes stepping stones, malicious attacker, cross virtual machine attack, virtual machine rollback etc [3].

2.2 Machine Learning- Machine learning is an application of AI which aims to build an automated model that could predict the further outcomes of a given problem by thoroughly getting trained with the historical data patterns that describes the problem [4,5]. The application of ML can be seen in the areas of prediction, image recognition, speech recognition, medical diagnoses, financial industries etc. ML algorithms can be categorized into three types:-

1. Supervised Learning- The data is labeled which is used for training the ML model. These models predict the outcome for the new input data by learning from the labeled data. Classification and Regression are the types of supervised learning. Random forest, Decision trees, logistic regression, support vector machine, naïve bayes, k-nearest neighbor are some of the supervised learning algorithms.

2. Unsupervised Learning:-The data is not labeled as well as not categorized which is used for training the ML model. This type of learning is very similar to the way human being learns. K-means clustering, DBSCAN, Principal Component Analysis, Mean shift algorithm are some of the unsupervised learning algorithm.

3. Reinforcement Learning:-In this learning model an agent tries to learn in its environment as it performs actions and observes the result of those actions in order to improve its performance. Markov Decision Process, Q learning are the learning models for reinforcement learning.

2.3 2.3 Genetic Algorithm:- Genetic algorithm lays its foundation from biological inspiration, in which the fittest or the best individuals are selected from the entire population of a generation and allowed to crossover in order to generate the next generation fittest population. This process is continued until the entire population is considered as the fittest [9]. There are five stages involved in genetic algorithm:-

1. Initialization:-Defining the population of individuals over which the algorithm is to be applied. Every individual is made up of single unit known as Genes (variable) which together in group are referred as chromosome (solution).

2. Fitness Function:-This function determines the fitness score of each solution of the population which in turns suggest that the given solution is an optimal solution or how close it is from the optimal solution of the defined problem.

3. Selection:-The individuals with best fitness score returned by fitness function are selected from the population and produce next generation solution. The various selection algorithms are Truncation selection, Tournament selection, Roulette wheel selection.

4. Crossover:-The selected individuals from the previous step

are allowed to exchange their genes by selecting crossover points (can be either one-point crossover or multi-point crossover). The various algorithms for crossover are Single-point crossover, Two-point crossover, Uniform crossover.

5. Mutation:-Some of the genes of the new offspring produced from the previous stage are changed in order to obtain the variety in the population. The algorithms for mutation are Bitwise Mutation, Gaussian Mutation.

2.4 Decision Tree- Decision Tree falls under the category of supervised learning which is used to solve both classification as well as regression problems. Internal node of the DT refers to the attributes whereas leaf node maps to the class labels. The selection of the root attribute is the major task in the construction of DT. This process of attribute selection can be accomplished by the calculation of Gini or Entropy values. The overview of DT algorithm can be summarized as, at each node the attribute selection is performed and split happens, for each split the purity metric is computed which depicts the gini or entropy value. The variable with lowest value of the purity metric is selected and this process is repeated until stopping criteria is met. The stopping criteria can be specific depth of the tree or number of value in the leaf node or specifying the minimum change in the purity metric from one split to another. The value of Entropy lies between 0 and 1 in contrast to Gini value that always lies between 0 and 0.5. ID3 (Iterative Dichotomiser 3), C4.5, CART (Classification & Regression Trees) are some of the popular DT algorithms. CART algorithm makes use of Gini, ID3 as well as C4.5 uses Entropy for calculating their impurity metrics. Gini can be calculated as:- $Gini = \sum_{i=1}^{n}(pi)2$ (1)

Entropy can be calculated as:-

$$Entropy = \sum_{i=1}^{c} - pi \log_2 pi \quad (2)$$

Where p_i denotes probability for classification.

2.5 Intrusion Detection Systems- The process of keeping an eye on the network and computer systems within the network for detecting any incident that may violate the integrity, availability and privacy is defined as intrusion and a system designed to detect such intrusions is termed as Intrusion Detection System. An intrusion detection system can be a software or a hardware or combination of both that can detect the occurrence of any of the type of intrusion and trigger warnings or alarms to the administrators for the taking the necessary steps in order to guard defence against such incidents. The presence of IDS in the cloud ecosystem plays a vital role in the process of identifying the occurrence of intrusions [6].

2.5.1 Intrusion Detection Approaches- Signature based detection and anomaly based detection are the two types of approaches used by an intrusion detection system [7].

Signature Based Detection or Misuse Detection or Knowledge Based Detection- The detection of known attacks, i.e. the attacks whose signatures or patterns are already saved in the database are only possible by this approach. The basic working of this type of IDS is that the alarms are triggered to the system administrators and attack logs are generated if the current activity finds mapping in the predefined attack signature database. The signature based approach offers a high degree of accuracy in terms of classifying only the known attacks and are thus not suitable to be used as a solo IDS for cloud environment.

Anomaly Based Detection- This type of IDS are used by the system to detect the unknown attacks including the zero-day attacks. The machine learning model or the knowledge-based or the statistical-based methods are applied for the purpose of detecting the intrusions by the anomaly based detection systems. The user activity or behavior is analyzed by any one of the approaches mentioned above in order to identify any abnormality in the user behavior triggers the alarms if any such activities are detected. There is a deal of time as well as computational complexity involved for the sake of detecting all the unknown potential threats from the current behavior of the user and can be considered as the limitations for such IDSs.

2.5.2 Types of IDS- Cloud based intrusion detection system (IDS) can be categorized into four types by the virtue of their deployment location within the cloud [8].

Host-based IDS- The Host-based IDS is primarily responsible for detecting intrusions on the particular host machine on which it is deployed such as virtual machine, hypervisor. HIDS triggers the alarm if any suspicious activity is detected by thoroughly observing the incoming and outgoing traffic from the deployed host machine.

Network-based IDS- Network traffic is monitored and analysed for detecting intrusions by the NIDS. The transport layer header as well as the IP of each and every data packets that penetrates the network is analyzed in order to detect the intrusions thereby providing classification between normal and malicious data packets. Both the approaches of intrusion detection, i.e. signature based and anomaly based intrusion detection are utilized by the NIDS.

VMM/Hypervisor IDS- The limitations of HIDS and NIDS are addressed by the HypIDS (hypervisor-based IDS) which are deployed at the hypervisor-level. This type of IDS is capable of monitoring the virtual machines, the host, virtual machines communications, hypervisor and virtual machine communications as well as incoming network traffic for the purpose of detecting any intrusion or malicious activities.

Collaborative/Distributed IDS- In order to monitor a large network, the utilization of distributed based IDS is done which comprises of several NIDS as well as HIDS deployed

over the entire network and within the network systems. Each of these deployed IDSs are responsible of detecting the intrusions at the network as well as at the host level either by signature or anomaly based approaches.

3 Related Works

Rabbani et al. [10] proposed a particle swarm optimization and probabilistic neural network based hybrid machine learning model in order to enhance the cloud security by accurately classifying the behavior of the user as normal or as malicious. PSO was used for optimizing the performance so that PNN can bring out accurate results in attack prediction. The performance metric of the proposed approach showed that all the nine types of attacks are accurately classified and higher true positive rate, precision as well as f-measure and low false positive rate is achieved. Subramanian et al. [11] proposed a research paper giving the highest emphasis on cloud security using CNN (Convolution Neural Network) machine learning algorithm. The main objective of the research is to perform analysis on the network traffic and detect abnormal activities using CNN-MSVM. The proposed approach makes use of MSVM (Multiple Support Vector Machine) instead of SVM in order to classify the various classes of the dataset. The proposed methodology achieves the detection accuracy of 98.87% for UNSW dataset and 98.6% for ISOT dataset and hence proving its superiority than other traditional approaches.

Hatef et al. [12] proposed a decision tree machine learning based hybrid approach for detecting the intrusions in the cloud environment for the sake of enhancing the cloud security. This hybrid system for detecting the intrusions works in four phases i.e., detecting the known attacks at the first stage, the second stage is responsible for detecting the known attacks that were not recognized by the first stage. This is followed by

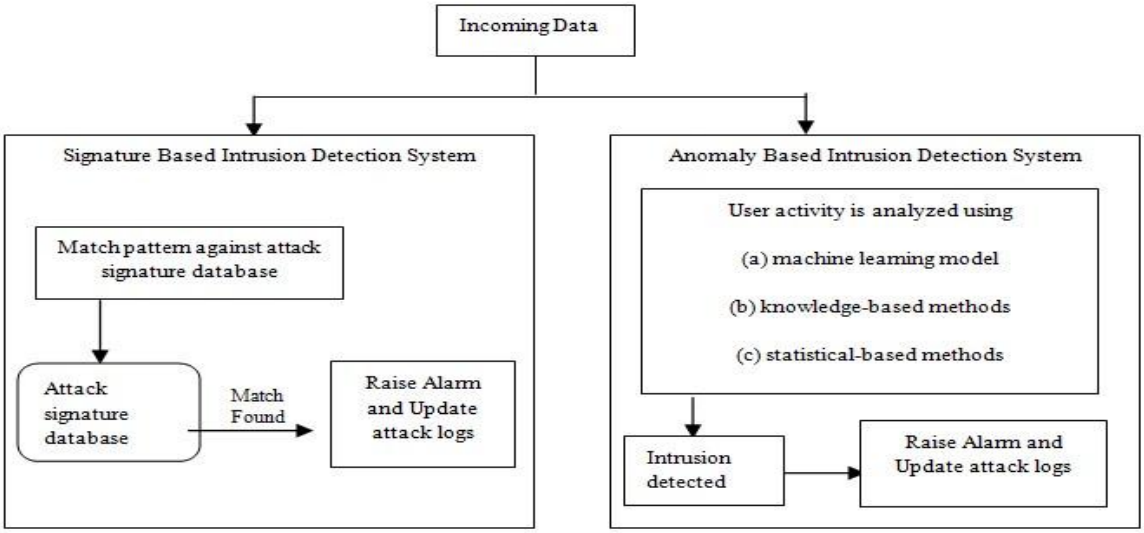


Figure. 1:-Intrusion Detection Approaches

updating of the known attack database with the attacks that bypassed the previous two detection stages and the final fourth stage takes care for creating the new attacks patterns by utilizing the knowledge of the previous known attacks. The simulation result clearly depicts that the proposed approach achieves increased rate in detection accuracy and types of intrusion detected.

The security of the robotic system based on cloud is suggested by the research paper of Gao et al. [13] which lays emphasis on enhanced NIDS (network intrusion detection system). The proposed approach makes use of semi-supervised FSSL-EL (fuzziness based semi-supervised via ensemble learning) mechanism. The proposed approach serves as a remedy for two of the major weakness of NIDS which either are trained by supervised or unsupervised machine learning algorithms and fails to achieve benchmark results in terms of increased intrusion accuracy and decreased false classification rate. An enhanced NIDS is proposed by Chiba et al. [14]. The proposed approach utilizes the power of optimized genetic algorithm as well as back propagation neural networks in order to achieve increased accuracy for NIDS. BPNN achieves high detection accuracy with minimum false classification due to the optimal values of the momentum and learning rate obtained by the proposed genetic algorithm which itself is optimized via fitness value hashing and parallel processing strategies.

Filho et al. [15] gave a framework for mitigation against DoS/DDoS cloud attack by using machine learning algorithm. The research lays emphasis on the detection of DoS and DDoS attack which are among the most dangerous threats of the recent era. CIC-DoS, CICIDS2017 and CSE-CIC-IDS2018 are the three benchmark datasets used to check the authenticity of the proposed methodology. The machine learning based detection system is superior than other approaches in several ways which includes the detection of other volumetric DDoS attacks that may overrun the network capacity are also made possible, secondly this system has no hardware or software requirements as well as no internal

examination of data packets is performed in order to maintain complete data privacy and also relevant feature are selected from the dataset by performing recursive feature elimination with cross validation. These features enable the approach to achieve a detection accuracy of more than 96% and reduced number of false classification. Meryem et al. [16] proposes a machine learning IDS. The approach is capable of detecting the unknown attacks having new signatures and if detected any such attack signature the approach reduces the false classification by training the ML model with the updated training dataset. Initially the proposed approach performs the structuring of the unstructured log files which contains the records of access to several files and the duplicate or useless log records thus found are removed by using the Map-Reduce. This is followed by labeling the unknown log records by K-Means algorithm and training the dataset with KNN, Support Vector Machine, Logistic Regression and Naïve Bayes algorithms. The proposed approach proves its superiority finally by updating the rules on detection of any new attack signature which helps in achieving the higher attack detection accuracy rate.

4 Proposed Approach

The proposed hybrid model is employed for the purpose of detecting the various cloud attacks from the incoming traffic well in advance before the malicious request reaches the cloud servers. This hybrid model is composed of genetic algorithm and decision tree classification algorithm (GA and CART). The hybrid model is trained on UNSW_NB15 dataset, which is considered as the authentic dataset containing nine different kinds of cloud attack [17]. Initially the baseline accuracy is calculated which includes all the features of the UNSW_NB15 dataset and is saved for the purpose of comparison. Then the genetic algorithm is called in order to produce the list of optimized individuals with selected features. The accuracy is calculated for each individual returned from the genetic algorithm and is finally compared with the initially calculated baseline accuracy. The most optimized features are then selected among the individuals with number of features as well as their accuracy score into consideration. The fitness function of the genetic algorithm calculates the accuracy score by making use of Pearson's correlation coefficient that indicates the degree of correlation among the features. The proposed work uses Pearson's coefficient in order to eliminate the features that are highly correlation with each other. The Pearson's correlation used in the fitness function of the genetic algorithm finds the highly correlated features in the dataset by taking the value of threshold as 0.6.

Pearson's correlation coefficient for sample is defined as

$$\boldsymbol{r_{xy} = \frac{S_{xy}}{S_x S_y}} \quad \text{....(3)}$$

where S_{xy} is the sample covariance and S_x, S_y are the sample standard deviations.

Pearson's correlation coefficient for population is defined as

$$\boldsymbol{P_{xy} = \frac{\sigma_{xy}}{\sigma_x \sigma_y}} \quad \text{....(4)}$$

where σ_{xy} is the population covariance and σ_x, σ_y population standard deviations.

Covariance is defined as $\boldsymbol{cov(x, y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n-1}}$ ....(5)

where x_i is the value of x, $\bar{x}$ is the mean of x, y_i is the value of y, $\bar{y}$ is the mean of y and n is the number of data points.

Genetic algorithm is used to determine the selected features for the machine learning model to predict the most accurate results, which in turns optimizes the input for the decision tree CART (Classification & Regression Trees) algorithm. The GA does the initialization of individual with 49 bits (total number of features in UNSW_NB15), 100 as maximum number of generation, with 0.95 and 0.1 as crossover rate and mutation rate respectively. This approach makes use CART Decision tree algorithm for classifying the incoming network traffic as malicious or normal in the second stage after receiving the optimized high score features from the genetic algorithm as input. The main reason for choosing CART DT algorithm as classifier is use of Gini by the CART algorithm for calculating the impurity metrics used for optimum feature split, in contrast with ID3 and C4.5 DT algorithms which makes uses of entropy for the same. Gini proves out to be more efficient than entropy as the calculation of entropy involves logarithms making it computationally heavy as well as slower than Gini calculation. And this in turn makes our proposed algorithm is computationally faster.

Algorithm 1: Hybrid Machine Learning and Genetic Algorithm based approach

Input : UNSW_NB15 dataset

1 Read the dataset UNSW_NB15
2 Divide the dataset into training and test sets
3 Calculate the baseline accuracy
  - 3.1 Create an individual with all the features present in the dataset
  - 3.2 Call the FitnessFunction with the individual created in Step 3.1 in order to obtain the baseline accuracy
4 Launch Genetic Algorithm, in order to obtain the optimized feature subset by running the genetic algorithm
  - 4.1 Initialization of individual
  - 4.2 Calculate fitness for individual created in step 4.1 by calling the Fitness Function
  - 4.3 Perform the selection operation over the individual created in step 4.1 using Tournament Selection
  - 4.4 Perform the crossover operation over the individual created in step 4.3 using Single Point Crossover
  - 4.5 Perform the mutation operation over the individual created in step 4.4 using Gaussian Mutation
  - 4.6 Check the stopping criteria of the genetic algorithm, i.e. if the maximum number of generations reached
  - 4.7 If the condition in step 4.6 is found true, then stop genetic algorithm and go to step 4.8 else return to step 4.1
  - 4.8 Return the list of all individuals found with optimized feature subset
5 Obtain the accuracy of each individual returned by step 4

	by calling the Fitness Function
6	Compare the calculated accuracy of each individual obtain in step 5 against the calculated baseline accuracy in step 3 and select the best feature subset
7	Create X_train and X_test dataset by including only optimized feature subset obtained in step 6
8	Train the Decision Tree CART Algorithm with X_train dataset obtained from step 7 for building the Machine Learning based Classification model
9	Test the model created in step 8 with test dataset
10	Calculate the performance metrics of the machine learning based classification model

Algorithm 2 : Fitness Function

Input :

individual :- binary vector of 0s and 1s

X_train :- training dataset of independent variables

X_test :- testing dataset of independent variables

y_train :- training dataset of dependent variables

y_test :- testing dataset of dependent variables

1	Traverse through the entire length of the individual and drop the features from the individual that is of no use, i.e. drop the features with value 0 from the X_train and X_test datasets
2	Apply Pearson Correlation on the X_train dataset in order to obtain the correlation between features
3	Identify those features that are correlated with other features with the threshold value of 0.6 (features which are highly correlated with threshold value of 60%)
4	Remove the highly correlated features found in step 3 from X_train
5	Remove the highly correlated features found in step 3 from X_test
6	Calculate the accuracy of X_train and y_train
7	Return the calculated accuracy obtained in step 6

5. Performance Matrix

In order to validate the results obtained by the classifier system, we used some of the below mentioned performance metrics parameters.

1. **Accuracy**- It is defined as the ratio between the summation of true positive and true negative to the summation of overall observation by the classifier.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}$$

2. **True Positive Rate** – Ratio between the true positive predictions to the total number of false negatives and true positives.

$$TPR = \frac{TP}{FN + TP}$$

3. **Precision** – The ratio of true positive to the summation of true positive and false positive results by the classifier is termed as precision.

$$Precision = \frac{TP}{TP + FP}$$

4. **False Positive Rate** – Ratio of positive prediction which were false to the total number of false positives and true negatives.

$$FPR = \frac{FP}{TN + FP}$$

5. **F1-Score**- The harmonic mean of recall and precision is referred as the F1-Score.

$$F1 - Score = \frac{2 * Recall * Precision}{Recall * Precision}$$

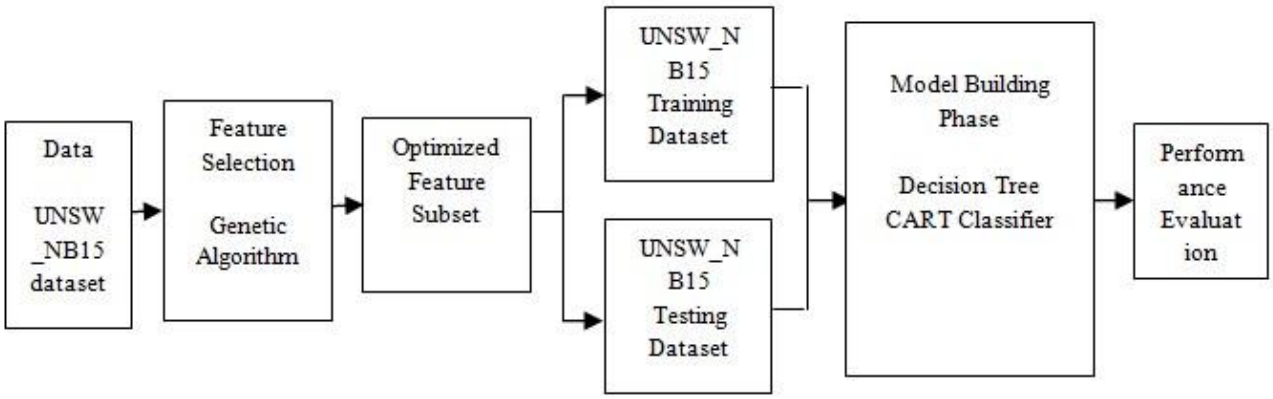


Figure. 2 Workflow of Proposed Approach

6 Environment Setup, Dataset and Results

The proposed work is implemented on Intel 11th Gen Intel(R) Core(TM) i5-1135G7 @2.40GHz machine having 8GB RAM and simulation is carried out in python using Jupyter Notebook. The proposed work utilizes UNSW-NB 15 dataset created by ACCS (Australian Centre for Cyber Security) and is considered as the authentic cloud attack dataset. This dataset is downloaded from kaggle in which the data is divided into four CSF files with 175,341 and 82,332 records in the training and testing datasets respectively. UNSW-NB 15 dataset has 49 number of features of nominal, integer, float, timestamp and binary data types. UNSW-NB 15 dataset contains total nine kinds of attacks categories namely Fuzzers (24,246 records), Analysis (2,677 records), DoS (16,353 records), Backdoors (2,329 records), Generic (215,481 records), Exploits (44,525 records), Shellcode (1,511 records), Worms (174 records), and Reconnaissance (13,987 records).

The genetic algorithm of the proposed approach produces the optimized number of the features containing only 12 features listed below in table 1 out of total 49 features that are of utmost importance for the machine learning classifier in order to produce accurate results. The dataset with the 12 optimized features are provided as input to the K-Nearest Neighbours, Naïve Bayes, Support Vector Machine, Decision Tree algorithms in order to construct the classifier and the results were compared. The result illustrated in table 2 against the above mentioned performance metrics, suggested that the Decision Tree CART classifier is best suited algorithm for the proposed hybrid model with 99.89% accuracy, 99.17% precision, 0.94% false positive rate and 98.98% true positive rate. The hybrid approach enables the machine learning model to overcome the major problem of overfitting, as the ML model is trained with the most optimized feature subset obtained from genetic algorithm. Additional the hybrid model is computationally faster as well as lightweight making it suitable for cloud based IoT devices.

S.No.	Selected Feature Name	Feature Description
1	Service	ftp, http, dns, ssh, smtp
2	Sbytes	Source to destination bytes
3	Sttl	Source to destination time to live
4	Dttl	Destination to source time to live
5	Sload	Source bits per second
6	Spkts	Source to destination packet count
7	ct_src_dport_ltm	No of connections of the same Source IP address and Destination port number in 100 connections
8	ct_srv_dst	No. of connections that contain the same service and destination IP address in 100 connections
9	ct_dst_src_ltm	No of connections of the same Source IP and Destination IP address in 100 connections
10	ct_dst_sport_ltm	No of connections of the same Destination IP address and the Source port number in 100 connections
11	ct_srv_src	No. of connections that contain the same service and source IP address in 100 connections
12	ct_src_ ltm	No. of connections of the same source IP address in 100 connections

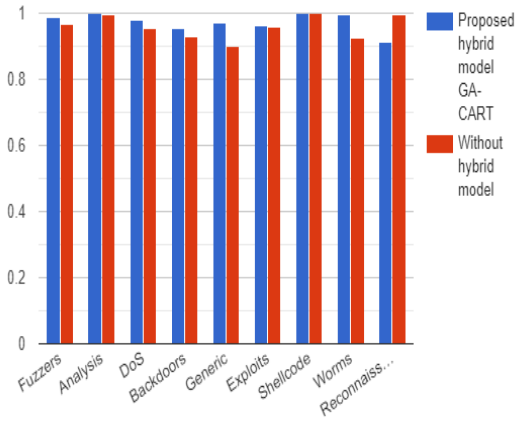


Figure. 3:-Accuracy

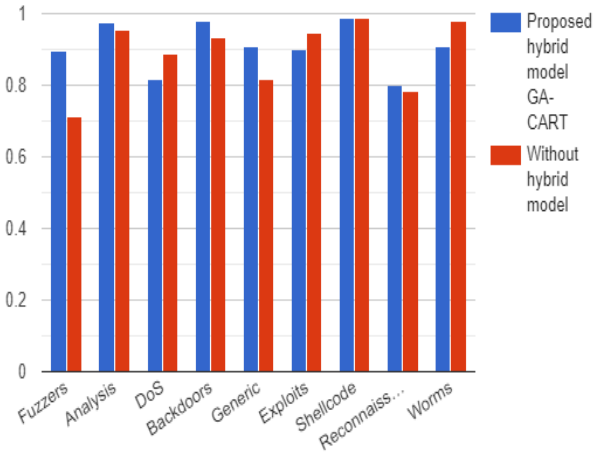


Figure.4:-True Positive Rate

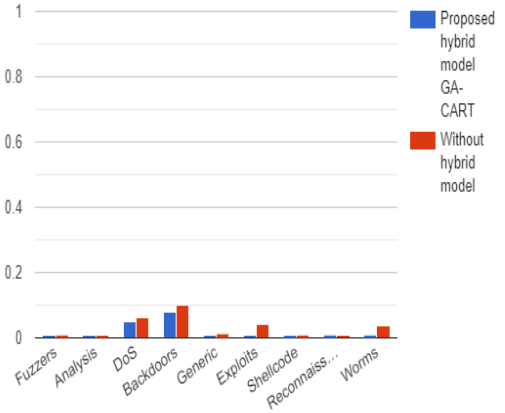
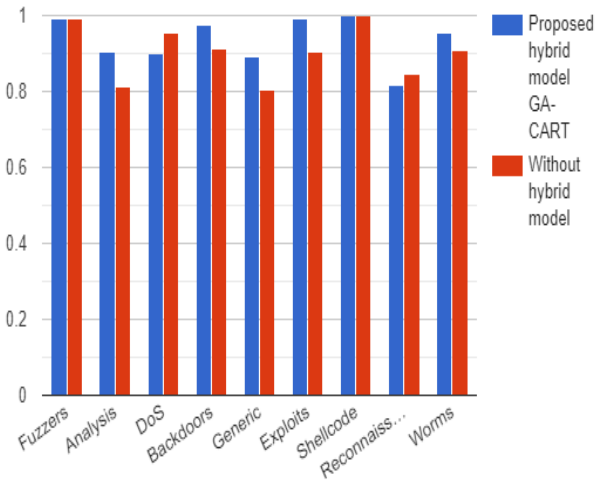


Figure.5:-False Positive



Rate Figure. 6:-Precision

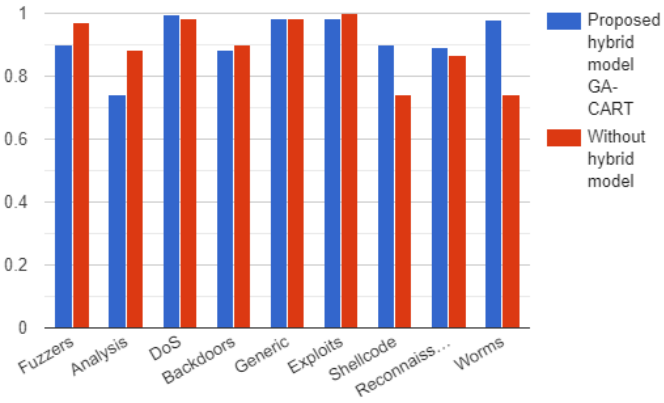


Figure. 7:-F1-Score

Table 2:-Performance Metrics results for nine types of attacks

%	Accuracy		True Positive Rate		False Positive Rate		Precision		F1-Score	
	Proposed hybrid model GA-CART	Without hybrid model	Proposed hybrid model GA-CART	Without hybrid model	Proposed hybrid model GA-CART	Without hybrid model	Proposed hybrid model GA-CART	Without hybrid model	Proposed hybrid model GA-CART	Without hybrid model
Fuzzers	98.89	96.70	89.38	71.44	0.7	0.88	99.17	99.11	90.05	97.12
Analysis	99.93	99.53	97.54	95.51	0.71	0.79	90.22	81.18	74.36	88.22
DoS	97.82	95.43	81.51	88.88	4.81	6.12	99.15	95.32	99.63	98.24
Backdoors	95.44	92.99	97.98	93.19	7.88	9.915	97.45	91.41	88.15	90.17
Generic	97.16	90.07	90.88	81.48	0.67	1.14	89.01	80.49	98.29	98.23
Exploits	96.19	95.88	89.83	94.79	0.69	3.98	99.14	99.30	98.21	99.87
Shellcode	99.89	99.92	98.86	98.78	0.79	0.87	99.81	99.95	90.10	74.27
Worms	99.49	92.45	80.09	78.18	0.89	0.75	81.82	84.48	89.37	86.51
Reconnaissance	91.39	99.49	90.81	97.99	0.87	3.791	95.47	90.75	98.11	74.24

7 Conclusion and Future Work

Cloud data security is of utmost importance in the current era. In this respect the proposed work introduces a hybrid model in terms of NIDS comprising of Genetic Algorithm and Decision Tree CART classifier. The GA is utilized for the purpose of feature extraction and enabling the dataset to get rid of redundant features, highly correlated features and having only the relevant features. This optimized feature subset allows the ML model to overcome the major problem of overfitting achieving the accuracy of 99.89%, 99.17% precision, 0.94% false positive rate and 98.98% true positive rate. Additional the hybrid model is computationally faster as well as lightweight making it suitable for cloud based IoT devices. In the future, we propose the hybrid model comprising Neural Network with Genetic Algorithm for network intrusion detection system.

References

[1] Bader Alouffi, Muhammad Hasnain, Abdullah Alharbi, Wael Alosaimi, Hashem Alyami and Muhammad Ayaz, A Systematic Literature Review

on Cloud Computing Security: Threats and Mitigation Strategies, IEEE Access, 2021, 57792 – 57807.

[2] Munish Saran, Rajan Kumar Yadav and Upendra Nath Tripathi, Machine Learning based Security for Cloud Computing: A Survey, International Journal of Applied Engineering Research (IJAER), Volume 17, Number 4, 2022, 337-336.

[3] Rajendra Kumar Dwivedi, Munish Saran and Rakesh Kumar, A Survey on Security over Sensor-Cloud, 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2019.

[4] Hongyu Liu and Bo Lang, Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey, Applied Sciences, 2019, 9(20).

[5] Umer Ahmed Butt, Muhammad Mehmood, Syed Bilal Hussain Shah, Rashid Amin, M. Waqas Shaukat, Syed Mohsan Raza, Doug Young Suh and Md. Jalil Piran, A Review of Machine Learning Algorithms for Cloud Computing Security, Electronics 2020, 9(9).

[6] Nathan Keegan, Soo‑Yeon Ji, Aastha Chaudhary, Claude Concolato, Byunggu Yu and Dong Hyun Jeong, A survey of cloud-based network intrusion detection analysis, Human-centric Computing and Information Sciences, Volume 6, 2016.

[7] Satyapal Singh, Mohan Kubendiran and Arun Kumar Sangaiah, A review on intrusion detection approaches in cloud security systems, International Journal Grid and Utility Computing, Vol. 10, No. 4, 2019.

[8] [Osama Alkadi, Nour Moustafa and Benjamin Turnbull, A Review of Intrusion Detection and Blockchain Applications in the Cloud: Approaches, Challenges and Solutions, IEEE Access, 2020.

[9] Sourabh Katoch, Sumit Singh Chauhan and Vijay Kumar, A review on genetic algorithm: past, present, and future, Multimedia Tools and Applications, Volume 80, 2020.

[10] Mahdi Rabbani, Yong Li Wang, Reza Khoshkangini, Hamed Jelodar, Ruxin Zhao and Peng Hu, A hybrid machine learning approach for malicious behaviour detection and recognition in cloud computing, Journal of Network and Computer Applications, Volume 151, 2020.

[11] E. K. Subramanian and Latha Tamilselvan, A focus on future cloud: machine learning-based cloud security, Service Oriented Computing and Applications, Volume 13, 2019.

[12] Mohammad Amin Hatef, Vahid Shaker, Mohammad Reza Jabbarpour, Jason Jung and Houman Zarrabi, HIDCC: A hybrid intrusion detection approach in cloud computing, Concurrency and Computation Practice and Experience, Volume 30, 2017.

[13] YING GAO, YU LIU, YAQIA JIN, JUEQUAN CHEN and HONGRUI WU, A Novel Semi-Supervised Learning Approach for Network Intrusion Detection on Cloud Based Robotic System, IEEE Access, Volume 6, 2018, 50927 – 50938.

[14] Zouhair Chiba, Noreddine Abghour, Khalid Moussaid, Amina El omri and Mohamed Rida, New Anomaly Network Intrusion Detection System in Cloud Environment Based on Optimized Back Propagation Neural Network Using Improved Genetic Algorithm, International Journal of Communication Networks and Information Security (IJCNIS), Volume 11, Number 1, 2019, 61-83.

[15] Francisco Sales de Lima Filho, Frederico A. F. Silveira, Agostinho de Medeiros Brito Junior, Genoveva Vargas-Solar and Luiz F. Silveira, Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning, Security and Communication Networks, Volume 2019, 2019.

[16] Amar Meryem and Bouabid EL Ouahidi, Hybrid intrusion detection system using machine learning, Network Security, Volume 2020, Issue 5, 2020, 8-19.

[17] Nour Moustafa, and Jill Slay, UNSW-NB15: A Comprehensive Data set for Network Intrusion Detection systems, Military Communications and Information Systems Conference (MilCIS), 2015.