Intrusion Detection using ML : A Survey

1814061
Tirth Thaker
IT Department KJSCE
Mumbai, India
tirth.thaker@somaiya.edu

1814062
Gopalkrishna Waja
IT Department KJSCE
Mumbai, India
gopalkrishna.w@somaiya.edu

1814061
Dhairya Umrania
IT Department KJSCE
Mumbai, India
dhairya.u@somaiya.edu

Abstract—**Intrusion detection is a vital aspect to any network based system. Often it is difficult to detect any intrusion until after it has happened. But every attack follows their own pattern and has their own set of attributes, which is why we can use machine learning algorithms to detect intrusions. In our literature survey paper, we look at various machine learning algorithms for intrusion detection. We have broadly classified machine learning algorithms into supervised, unsupervised and ensemble learning algorithms and compared their performance in detecting intrusions. For supervised learning, we have included several algorithms such as: KNN, SVM, Logistic regression, naive bayes classifier. For unsupervised learning we looked at algorithms such as: k-means, isolation forest, EM, SOM. For ensemble learning we have looked at a diverse range of boosting, bagging and stacking algorithms. We have also given a dataset description of each dataset that has been covered for any of the algorithms included in the paper.**

Index Terms—**Intrusion, IDS, anomaly, supervised learning, unsupervised learning, ensemble learning.**

I. INTRODUCTION

An Intrusion in the realm of network security broadly refers to any unauthorized or illicit activity on the network. In today's contemporary world, a network's security is plagued by intrusion attacks like Dos, brute force or attacks from inside a network which lead to compromised systems, theft of data and even huge financial losses. An intruder is an attacker or a system with limited or no access rights who tries to obtain more authority or has an intention of stealing or damaging confidential data from the victim's system. These attackers seldom leverage network vulnerabilities like bugs, frail security-policies, or network design flaws to intrude into the network. An Intrusion Detection System (IDS) is used to monitor network activity for detection and identification of policy violations, threat or malicious activity.

Detection techniques can either be signature-based or anomaly-based. In signature detection, the IDS compares the incoming packets with the unique signature or pattern of known attacks present in a database. Almost all the traditional IDS developed were Signature-based systems, which means they used pre-defined signatures of attacks and configurations for detection of malicious activities, however this technique has a major drawback.The database containing the attack signatures need to be frequently updated since the signature based IDS will not be able to identity an attack unless it has the signature of the attack in its database and hence It it not able to identify new attacks also known as zero-day attacks. The arrival of machine learning helped to overcome this limitation of the signature based systems by leading to the development of anomaly detection systems. In anomaly detection technique the IDS uses statistical quantities to define the thresholds of different parameters for normal activity and any deviations from these normal thresholds can indicate the occurrence of a malicious activity. These systems use ML to train a model to simulate normal network activity and then compare the behaviour of new examples with the existing model. If the event shows any deviation from the normal network activities then the system flags the event as anomalous and indicates that the event can be an intrusion.

Over the past decade researchers have experimented with plenty of ML techniques with the aim of augmenting the detection-rate, minimizing false positives and maximizing the accuracy of IDS. These techniques can be categorised as Supervised Learning techniques, Unsupervised Learning techniques and Ensemble learning techniques.

In this survey paper we have surveyed different papers exploring the aforementioned techniques and contrasted them based on the accuracy of the models trained on different datasets. The paper is arranged as follows: Section II gives the description of the various datasets utilized in the different papers. Following this the next three sections i.e Section III, IV and V deal with the survey of different papers supervised, unsupervised and ensemble learning. Finally this paper ends with a discussion of the results and a conclusion.

II. DATASET DESCRIPTION

Dataset is a collection of records used to build a ML model. Each instance of the dataset consists of many features or dimensions, these features are also known as the attributes of the instance. Below we have listed and described the datasets which were used in the papers we have surveyed.

1) KDD-1999 Cup dataset: This dataset was provided during the 3rd International Knowledge Discovery and Data Mining Tools Competition and it has standard data on a gamut of intrusions simulated in a military network environment. The dataset has 4,898,431 network-traffic training instances out of which 1,074,992 are unique and 311,027 network-traffic test instances out of wich

TABLE I
DATASET DESCRIPTION

	Dataset Statistics			
Name	**Training**	**Testing**	**Features**	**Labels**
KDD-1999 Cup	1,074,992	311,027	41	5
NSL-KDD	125,973	22,544	43	5
UNSW-NB15	175,341	82,332	42	10
CICIDS2017	2,830,108		78	79

77,289 are unique with each instance having 41 features and one of the 5 labels: DOS, R2L, U2R and probing. A DoS attack leads to denial of services to legitimate users. Probe attacker is the type of attack in which the attacker collects sensitive information and fingerprints of the target machine which can be used to plan future attacks. In R2L, the attacker's aim is to obtain unauthorized access to the target system while in U2R, the attacker gains access to an unauthorized machine which is then used to obtain root access.

2) Canadian network traffic NSL-KDD dataset: This dataset is an enhancement of the KDD-CUP 1999 dataset and is another standard dataset which is used for experimental analysis and training models for an Intrusion detection system. The dataset has 125,973 network-traffic training instances and 22,544 network-traffic test instances with each instance having 43 features. The instances are labeled from 1 of the following 5 categories: Normal/benign, Probe, Remote to Local (R2L), User to Root (U2R) and Denial of service (DoS). The NSL-KDD dataset is better than the KDD99 dataset as unlike KDD99 the NSL-KDD it does not have duplicate instances which avoids the model to bias towards the more frequent examples and also the size of the dataset is much more reasonable when compared to the KDD99 dataset which makes it easier to experiment on the entire dataset rather than use sampling.
3) UNSW-NB15: It was created by Cyber Range Lab of the Australian Site of Cyber Security. The instances mentioned in the dataset have 42 features (excluding the labels) and are categorized in one of the ten attack classes, also it contains a feature indicating an instance of an attack or normal flow (can be used for binary classification). It contains around 175,341 examples in a training set and 82,332 in a test set which is comparable to the NSL-KDD dataset.
4) CICIDS2017: This dataset simulates real world network and uses CICFlowmeter V3.0 to extract 78 features ,79 labels and 2,830,108 instances.It has attack like brute force FTP, brute force SSH,DDoS attacks.It consists of data simulated from multi-stage attack scenarios. Since this dataset is so comprehensive most of the researchers use a subpart of this dataset called Machine-Learning.CSV data, which consists of only 14 types of attacks. part of the CICIDS-2017

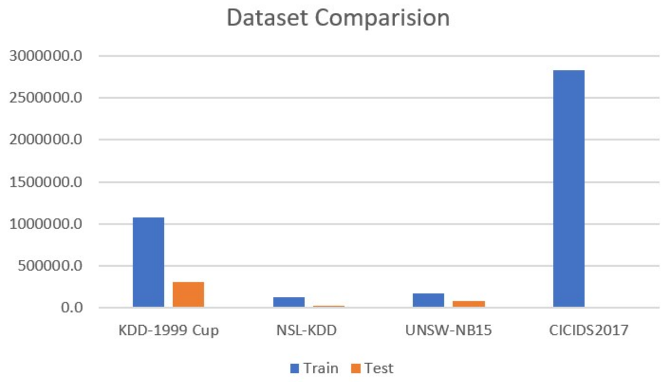

Fig. 1. Dataset comparision

III. SUPERVISED LEARNING

In Supervised learning intrusion detects labeled training data.Supervised learning consist of two stages mainly test and training dataset. Training stage has relevant features and classes are identified and the algorithm learns from these data samples. In supervised learning each record is a pair, containing a network or host data source and an associated output value ,namely intrusion. A supervised learning algorithm is then used to train a classifier to learn the relationship that exists between the input data and the labelled output value. Classification method in supervised learning are decision trees, rule-based systems, neural networks, support vector machines, Naive Bayes and nearest-neighbor In paper [1] the authors have used the UNSW-NB 15 dataset for binary classification. they have compared the effect of two supervised learning algorithms for binary classification of intrusion detection. Before training the model on the train dataset normalization was performed and then 18 features out of 42 were selected using feature selection technique. The effect of the following 2 models was compared:-

1) KNeighbors (KNN) :A nonparametric simple learning algorithm KNN use to classify based on the majority of the nearest neighbours calculated using similarity or distance metrics like Euclidean,Manhattan.
2) Support Vector Machine:It is a supervised machine learning algorithm used in binary classification and regression. It maximizes the margin around the separation hyperplane. Support Vectors used to minimal subset of the training observations and are used as base for the optimal location of the decision hyperplane..

To determine the performance measures of the classifiers a 2-by-2 confusion matrix was calculated which provide Accuracy metric. The results showed the KNN algorithm had accuracy of 89.4 % and an AUC score of 0.97 which meant that the classifier was able to provide a good separability between the positive and the negative class. The SVM algorithm gave an accuracy of 92%.A UC score of SVM was also 0.97 which indicates that this classifier also provided high separability. As a Future Work they proposed that the performance of these algorithms should be tested on other dataset like the UNSW 2018 IOT Botnet dataset.

In paper[2] the authors have made use of Decision Tree, Random Forest,Multinomial Logistic Regression and Multinomial Naive Bayes as the supervised learning classifier. The authors have used the UNSW-NB 15 dataset for binary classification. First they have performed pre-processing and normalization on the input dataset and then feature reduction is done by using a feature variance threshold. After feature reduction the processed dataset is then trained on the 4 models mentioned below and their performance is calculated.

1) Random Forest Classifier: This algorithm uses feature subspace to the model.It convert dataset into a bootstrap dataset and then aggregates the final model.It is aggregate to produce a final prediction for the Random Forest algorithm.Accuracy of 0.85was received for Random Forest.
2) Decision Tree: Decision tree is constructed where the parent node is selected based on max entropy values. Splitting of nodes is done based on entropy values of other parameters. Depth of a given tree can be stopped when no further change in the entropy values . They used the C4.5 Decision Tree. Accuracy of 0.86 was achieved using a Decision Tree
3) Multinomial Logistic Regression: It used a cross entropy loss function. This algorithm is used when more than two classes are present.Accuracy of 0.76 when the multinomial option is set to Newton cg.
   Naive Bayes:It is used for classification based on the probability of having an instance as an attack. IBayes Theorem whereby features are assumed to be conditionally independent of each other was applied on it. The accuracy achieved was 0.71

We see that the Decision Tree has a better accuracy of 0.86 among all four. It is able to reduce the number of features from 49 to 37 using the novel Variance Threshold method. The results obtained are encouraging and in the future distributed Machine.Learning algorithms can be applied to do the faster computation for large datasets.

IV. UN-SUPERVISED LEARNING

nsupervised learning is a subset of machine learning algorithms where the inferences are made from the input dataset and not from the tagged labels, or predefined output. In unsupervised learning, the algorithms use techniques such as clustering to group the data and draw inferences from them. It uses the features of each tuple to check how similar they are and groups the most similar ones together. The similarity is based on the attributes and features the dataset possesses.[1] A few examples of unsupervised learning that we will have a look at in this paper are: K-means clustering, isolation forest, Expectation Maximization and Self-organizing map. K-means clustering finds out how many clusters are optimal for a particular dataset based on the euclidean distance of each point in a cluster to its center. It is a widely used parameter in Wireless Sensor Networks (WSN). Isolation forest algorithm is used to find out anomalous behaviour and it is used to detect outliers in large datasets. It consists of random tree forests which have been derived from the different attributes and a randomly selected value in the range of the respective attribute. This forms a tree structure and the length to the path from the root node to the end node is what finds the anomalies [1]. Expectation maximization, EM, is similar to k-means but it calculates the probability of each cluster based on probability distributions and maximizes the probability of the entire data based on the clusters formed. Self-organizing map is sort of an unsupervised neural network, where it forms clusters based on inputs which have similar effects on the neurons and the underlying network[3].

In paper [1], Portela, Mendoza and Benavides have used K-means clustering to find out the different types of attacks we can identify from the dataset. The dataset used is the UNSW-NB 15 dataset, and the features used from this dataset for clustering are: ’smean’, ’ct srv src’ and ’dloss’. Using these three parameters, the optimal number of clusters was found out to be 5, based on the elbow method. The distance measure used was Euclidean distance. After running the algorithm, they found 5 clusters as shown in fig. 1. From the 5 clusters, 2 of the clusters, yellow and red, indicate normal behaviour. Brown clusters were identified as exploit attacks. Green cluster group was identified as attacks belonging to a category known as Fuzzers and the purple cluster group indicates Denial of Service attacks.

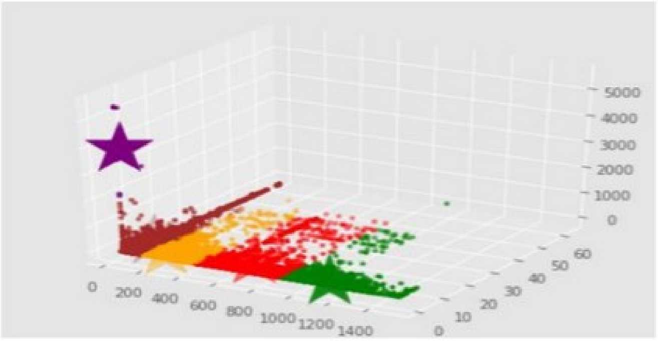

Fig. 2. K-means Clusters formed in paper [1]

Portela, Mendoza and Benavides have also used the isolation forest algorithm in paper [1] to detect intrusion based on anomaly detection. Once again the UNSW-NB15 dataset was used for this algorithm. The parameters they chose were as follows:

- Number of trees in the forest = 100

TABLE II
SUMMARY OF THE SURVEY

Paper	Algorithms	Dataset	Overall Accuracy	Comments
[1]	The SVM	UNSW-NB15	92%.	This approach gave a high separability indicated by AUC score of 0.97
[1]	Isolation Forest	UNSW-NB15	45%	Low success rate and it means that it was difficult to isolate the outliers in this dataset
[2]	Decision Trees	UNSW-NB15	86.12%.	This paper suggested to test the model on other dataset set to check it generalizability.
[3]	EM Classifier	CICIDS2017	60.06%	It had a poor detection rate
[4]	Stacking ensemble of DT, SVC and LR	NSL-KDD	84.1%	Low detection rate for R2L and U2R with 1% and 16% respectively
[5]	Ensemble of DT, DNN, Multitree, RF, KNN models	NSL-KDD	85.2%	Improved detection rate for R2L and U2R with 55.27% and 25% respectively
[6]	Random forest with SMOTE+ENN	KDD-1999	99.2%	Drastic improvement in detection rate of R2L and U2R with 99.9953% and 99.997% respectively

- Number of samples extracted to estimate each tree = 256
- The default value of contamination in the dataset = 0.22
- The number of features to extract from the dataset in training phase of each estimator = 1.0
- Sampling was performed without replacement
- Processors were to used parallely
- Random number generator seed = 42

With all these parameters the algorithm had an accuracy of 0.45. AUC curve displayed a result of 0.49. This indicates a low success rate and it means that it was difficult to isolate the outliers in this dataset hence having a low probability to predict intrusions.

Authors in [3] have used the CICIDS2017 testing dataset for their unsupervised machine learning models: k-means, EM and SOM. Their k-means clustering model found out the best number of clusters to be 4, and maximum iterations to be 300. This model achieved the following results: accuracy = 23.41%, precision = 67.37%, recall = 23.41%, F1-Score = 37.36%. The EM classifier also got 4 clusters but had a poor detection rate. The EM model got the following results: accuracy = 60.06%, precision = 86.88%, recall = 60.06% and F1-Score = 74.11%.

The SOM algorithm had a poor attack detection ability and has a higher number of false alarms than other algorithms. The results for SOM are as follows: accuracy = 59.06%, precision = 85.88%, recall = 60.00%, F1-Score = 74.11%.

V. ENSEMBLE LEARNING

Ensemble Learning is a type of supervised learning technique in which a combination of ML models known as weak learners is used to create a strong learner, where individual decisions of the weak learners are combined to get to the consensus. It works on the principle that a combination of multiple ML algorithms can be leveraged to gain enhanced predictive performance than any of the constituent individual algorithms alone. In the papers we have studied a diverse variety of ensemble methods like Boosting, Bagging and Stacking have been used for Intrusion detection. Boosting is a type of ensemble technique in which a homogeneous set of weak learners are trained sequentially on a dataset which is re-weighted according to the misclassifications, example for this type of technique is AdaBoost which uses decision trees as weak learners. Bagging refers to parallel training of homogeneous weak learners on different subsets of the same dataset, Random Forest is an example of Bagging algorithm which uses an ensemble of decision trees. Finally stacking trains multiple heterogeneous weak learners on the same dataset. The merit of ensemble learning is that it improves generalizability and robustness over a single model.

In paper [4] Masud and Mustafa have proposed a voting based ensemble learning approach for network intrusion detection. Here they have compared the performance of individual base learners with 5 different Stacking ensembles created by using different permutations of Decision trees, SVC, Logistic regression and gradient boost algorithms and have made use of the NSL-KDD dataset. First they have performed some preprocessing on the dataset like removal of instances with missing values, one hot encoding the categorical features and dropping features with constant value and then they have used this dataset to train the individual base classifiers and the 5 ensembles which have used max voting for predicting the output. Using recall as the evaluation metric they found that the ensemble learners outperformed the individual base learners and the best result of 84.1% accuracy was obtained for the ensemble using Decision Tree, KNN and Logistic regression as the base learners. Also analysis of recall of individual class showed that though the performers of this ensemble methods was high for the normal ,DoS and probe classes they were not able to identify the examples instances belong to R2L and U2R classes and had a recall of 1% and 16% respectively and they suggested that future work should be done to improve this.

X.Gao and C.Shan in paper [5] have trained several ML models like KNN, Decision trees, random forest, KNN, logistic regression, SVM, DNN and multi-tree on the NSL-KDD train dataset and then they have created ensemble of 5 model selected 5 models giving the best results to be on the cross-validation test and here instead of using hard voting, they have used weighted voting where the weights to the base learners were given based on the results of the cross-validation. The result of including sophisticated models like DNN, Multitree and Random forest and using a weighted voting was that there was a significant increase in the performance measures than those obtained in [4] with R2L and U2R classes giving a recall of 55.27% and 25% respectively. Though this improvement in performance was quite remarkable the authors suggested that the performance can be further improved by using some class imbalance reduction technique.

Class imbalance refers to a situation, where some classes are highly underrepresented compared to other classes making it difficult for many conventional ML algorithms to predict effectively the examples belonging to the minority class. As per the suggestions of the paper [5], the authors in [6] have used SMOTE+ENN imbalance reduction technique in combination with Random forest which is a bagging based ensemble on the KDD-1999 dataset with 25 features present after feature selection based on variance to enhance the performance of models to classify R2L and U2R attacks. Synthetic Minority Over Sampling technique (SMOTE) is an oversampling technique which involves creation of 'synthetic' samples from existing minority instances by finding the K-neighbours of the instance and then randomly selecting a vector point between the current point and the K neighbours. Edited Nearest Neighbour is an undersampling technique in which for each instance in the majority class if the majority class of the instances K-nearest neighbor and its label is different, then the instance and its K-nearest neighbor are deleted from the dataset. Following this methodology the authors received a dramatic increase in the recall for classification of R2L and U2R for 55.27% to 99.9953% and 25% to 99.997% respectively.

VI. Discussion and Future work

VII. Conclusion

References

[1] F. G. Portela, F. Almenares Mendoza and L. C. Benavides, "Evaluation of the performance of supervised and unsupervised Machine learning techniques for intrusion detection," 2019 IEEE International Conference on Applied Science and Advanced Technology (iCASAT), 2019, pp. 1-8, 2019.

[2] A. Srivastava, A. Agarwal and G. Kaur, "Novel Machine Learning Technique for Intrusion Detection in Recent Network-based Attacks," 2019 4th International Conference on Information Systems and Computer Networks (ISCON), 2019, pp. 524-528, 2019.

[3] Z. K. Maseer, R. Yusof, N. Bahaman, S. A. Mostafa and C. F. M. Foozy, "Benchmarking of Machine Learning for Anomaly Based Intrusion Detection Systems in the CICIDS2017 Dataset," in IEEE Access, vol. 9, pp. 22351-22370, 2021.

[4] M. Raihan-Al-Masud and H. A. Mustafa, "Network Intrusion Detection System Using Voting Ensemble Machine Learning," 2019 IEEE International Conference on Telecommunications and Photonics (ICTP), pp. 1-4, 2019.

[5] X. Gao, C. Shan, C. Hu, Z. Niu and Z. Liu, "An Adaptive Ensemble Machine Learning Model for Intrusion Detection," in IEEE Access, vol. 7, pp. 82512-82521, 2019.

[6] T. Lu, Y. Huang, W. Zhao and J. Zhang, "The Metering Automation System based Intrusion Detection Using Random Forest Classifier with SMOTE+ENN," 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT) pp. 370-374, 2019.

[7] U. S. Musa, M. Chhabra, A. Ali and M. Kaur, "Intrusion Detection System using Machine Learning Techniques: A Review," 2020 International Conference on Smart Electronics and Communication (ICOSEC), pp. 149-155, 2020.

[8] U. S. Musa, S. Chakraborty, M. M. Abdullahi and T. Maini, "A Review on Intrusion Detection System using Machine Learning Techniques," 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), pp. 541-549, 2021.