An Intellectual Detection System for Intrusions based on Collaborative Machine Learning

Dhikhi T[1]
Research Scholar, Saveetha School of Engineering
Saveetha Institute of Medical and Technical Sciences
Assistant Professor, SRM Institute of Science and Technology, India

Dr. M.S. Saravanan[2]
Professor, Saveetha School of Engineering
Saveetha Institute of Medical and Technical Sciences
India

***Abstract*—The necessity for safety of information in a network has inflated due to the impressive growth of web applications. Several methods of intrusion detection are used to detect irregularities which depend on precision, detection frequency, other parameters and are anticipated to familiarize to vigorously varying risk scenes. To accomplish consistent abnormalities detection in a network many machine learning algorithms have been formulated by researchers. A technique based on unsupervised machine learning that use two separate machine learning algorithms to identify anomalies in a network viz convolutional autoencoder and softmax classifier is proposed. These profound models were skilled as well as evaluated on NSLKDD test data sets on the NSLKDD training dataset. Using well-known classification metrics such as accuracy, precision and recall, these machine learning models were assessed. The developed intrusion detection system model experimental findings showed promising outcomes in anomaly detection systems for real-world implementation and is compared with the prevailing definitive machine learning techniques. This strategy increases the detection of network intrusion and offers a renewed intrusion detection study method.**

Keywords*—Intrusion detection; machine learning; deep learning; convolutional autoencoder; softmax classifier; NSL-KDD dataset

I. Introduction

Detection of intrusion is to track device or network anomalies. This analyze both known and unknown attacks. Many approaches are processed to find the anomalies. As information is valuable resource the security is a crucial thing, thereby small complication intrusion detection system is a demanding assignment. Detection of intrusion detects external intrusions and monitors unauthorized inner user operations by recognizing and reacting to malicious network communication and computer utilization behaviour. IDSs plays an active role in network monitoring and was usually used as a network security element in latest years. Moreover, aims to detect intrusions by studying the process and features of intrusion conduct, thus allowing a process of invasion Two vital intrusion detection technologies be present, viz, anomaly detection and misuse detection [1]. Intrusion detection systems are classified as two: Network IDS, Host IDS [20]. The source of information accommodates IDS audit information. IDS triggers alarm by evaluating that audit information as it detects intrusion or attack [23].

Feature selection specifies the selection of the appropriate feature subset from extra dimensional quality depend on different calculation parameters, thereby achieving a model. In this research methods based on machine learning for intrusion detection is focused on [15]. Machine learning methods can be classified into i) supervised techniques ii) semi supervised learning iii) unsupervised learning methods. In this research multiple supervised learning methods for IDS is explored with regard to their performance metrics viz false alarm rate (FAR), accuracy, recall, F1 measure, time taken to train and test each classifier. NSLKDD database includes only selected dataset records that furnish a great research of different intrusion perception method for machine learning. NSLKDD incorporate 41 input together with class names [19]. In addition, the archive in the NSLKDD trained and tested sets is fair. This strength makes it inexpensive to execute the entire set of research without randomly selecting a small part. Accordingly, the assessment aggregation of various study job is coherent as well as similar. This excludes repetitive train records, because classifiers are not prone to ever-increasing records.

Rest of the paper is separated into five sections. Section II presents salient works associated to IDS. Section III offers the planned framework of Convolutional Softmax IDS and mention the different steps involved in the model. Section IV discusses on the evaluation criteria of the performance. Section V examines on the outcomes of the research along with comparison of results. Followed by Section VI describes the conclusion and next presents the references.

II. Existing Work

R. Vinayakumar [1] tells a profound DNN, a sort of deep learning model, is being studied for creating flexile and efficient IDS to categorize unanticipated and uncertain cyber-attacks. This sort of research promotes the identification of the finest algorithm that can operate efficiently to detect potential cyber-attacks. On several publicly accessible benchmark malware databases, a thorough analysis of DNN experiments and other classical machine learning classifiers is shown. It is confirmed by strict experimental testing that DNNs compared with classical machine learning classifiers, perform well. Shone [6] provides an original methodology of deep learning to detect interruptions that demonstrates that deep learning grouping is build using stacked NDAEs. This was used to evaluate the expenditure of the normal KDD Cup and

NSLKDD datasets in graphics processing unit enabled Tensor Flow. Moreover, measured the preparation time required for the stacked NDAE model, as well as a DBN model to examine the KDD99 dataset furnishing large accuracy. The well-known methods of machine learning were evaluated by I. Ahmad et al [10]. Support mechanism of vectors and machine of extreme learning. To evaluate the interruption detection system, the NSL datasets are considered. It is found in their assessment result that ELM is enhanced accurate. Al-Qatf [8] suggested a STL IDS approach that is effective in-depth training for learning features and dimensionality employing auto encoder machine to restructure an illustration of a novel function in an unsubstantiated way.

Naseer [12] explored the appropriate anomaly-based strategy to IDS produced on multiple profound ANN like convolutionary neural, regular neural systems and auto encoders which are competent on NSLKDD dataset. These are done on a GPU-related test bed that uses theano-backed keras. Evaluations were conducted using metrics of the organisation viz. Receiver operating attribute, curved region, accurate curve, mean average accuracy, conventional ML technique classification. M. H. Ali [13] implemented a Fast Learning Network knowledge model to support particle swarm optimization (PSO). This is useful in identifying entrant and KDD99 data set is endorsed. The scheme developed is associated with a nice variety of meta-heuristic schemes for tutoring both the extreme learning scheme and the FLN classification scheme. Within the testing precision of the training, PSO-FLN has defeated various teaching methods. P. Tao [14] recommends fresh inherited operation hinge on the features of GA and SVM algorithms, FWP-SVM. The stated technique reduces rate of SVM mistake that use a genetic algorithm option approach to modify the fitness algorithm. SVM's distinctive weights and limitations are simultaneously optimized, enabling optimum subset of features. The result of this article defines the right favorable rate of escalation and decreases the velocity of mistake. Q. Zhang [15] utilized fuzzy depends on the kernel – a set of KDD 99 data set for IDS validation and analysis. These blurred classifiers operate upon discrete, noise data's inaccuracy and vagueness, thus performing well in terms of effect and accuracy in reduction. The function selection techniques were commonly in use laterally with classifiers for network interruption identification.

H. Peng [16] exploited improved choice of features, FACO merged ant colony optimization algorithm for set of features. To improve the cataloging of separate classifiers, FACO is introduced. This optimization algorithm is an algorithm for simulation optimization that creates a detailed directed graph over n features, imitating ants ' scavenging behaviour. In addition, excess features are allocated to reduce the instance difficulty in grouping algorithms as well as enhance traffic allocation efficiency. Z. Wang [18] article assess different algorithms for intrusion catching domains using deep learning approaches and define different element application models for attack algorithms. Research indicates that the most commonly used highlights show their greater contribution to the exposure of intrusion detection created by the intense understanding and thus warrant additional consideration. Nisioti[17] provides comprehensive overview of an unattended and a crossed disturbance recognition approaches, examining their spatial potential. It characterizes the importance of highlighting construction techniques and also discuss actual IDS's should progress connection and attribution of the fundamental place. Moreover, suggested three innovative components related to communication on the outbound network. Haipeng Yao [4] introduces a multilevel model for IDS called multilevel semi-supervised ML (MSML). A notion of "pure cluster" is implemented in the module and implemented a semi-supervised hierarchical k-means algorithm. The "unknown pattern" and cluster-based technique is described in pattern discovery module. The updating module model offers a retraining mechanism. To evaluate MSML, the KDDCUP99 dataset is used. Experimental findings indicate that MSML is superior corresponding to general precision, F1-score, and unknown pattern recognition capacity to other current intrusion detection models.

III. Proposed Research

The system is planned to corporate trust unsupervised machine learning algorithms to boost the system's accuracy and efficiency. This model as shown in Fig. 1 compromises of distinct stages preprocessing, normalizing data, unifying data, feature extraction, classification, training and testing dataset. Two machine learning algorithms are implemented for training and testing dataset. A combination of both algorithms is also implemented for train and test datasets that improves the performance parameters [5]. The proposed approach uses collaborative supervised algorithms that offer an effective deep learning method. Thus, advances the performance of the model associated to the prevailing methods. This scheme combines convolutional autoencoder and softmax classifier for feature extraction and classification, respectively.

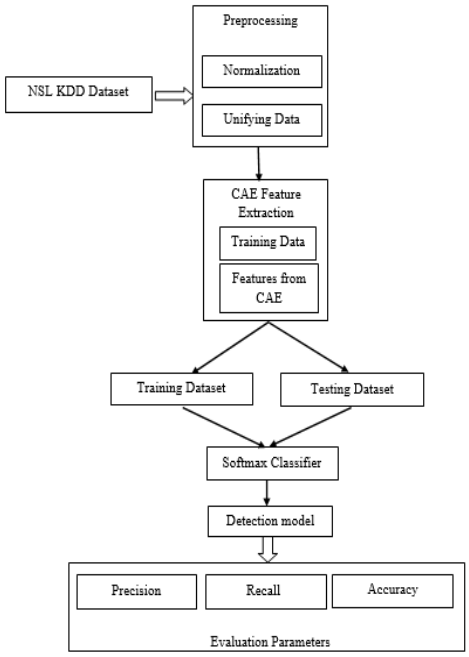

Fig. 1. Proposed Framework.

Preprocessing: The system's achievement is directly in proportion to the data set's accuracy, so the collection of data is a significant task. KDD 99[22] is utilized for anomaly detection valuation, occupies specific inspection data that consider a broad range of simulated intrusions. NSL-KDD [7] is a data set that is planned to resolve some of KDD99's key difficulties. The attacks in NSLKDD are classified mainly into four types as in Table I [11]. The protocol types in the dataset are shown in the Table II.

Preprocessing can be performed to remove symbolic characteristics in the procedure of identification. These types of symbolic data cannot be processed by the classifier to improve the efficiency of detection advancement. Pre-processing phase minimize data to a great extent as possible without loss of information and requires specific scheduling, preparation and testing. This helps to provide IDS with appropriate and effective information on computation, filtering fake rates and improving detection rate and to find patterns of attack and show suitable kinds of information for policy making by administrators.

In our approach the non -numeric values are changed to numerical values. Every attribute in the dataset are transformed into numeric values. In preprocessing, normalization of data is done. The intention of normalization is to alter numeric column values in the dataset to a popular scale without distorting value range distinctions [2]. Every dataset does not involve standardization for machine learning. It is only needed when there are distinct ranges of characteristics. Numerous NSL-KDD dataset features have wide ranges of maximum to minimum value, with a maximum value 58,329 also a minimum value 0. These features of dataset are normalized using min-max normalization and thereby maps the range from 0 to 1 using the equation (1)

$$v' = (v - min_F / max_F - min_F)\ (new\_max_F - new\_min_F) + new\_min_F \quad (1)$$

where v denotes the data point, min_F is the minimal value for all data and max_F is the upper limit value for all data factors. new_ min_F and new_max_F are the newly mapped minimum and maximum value, respectively.

Feature extraction: Feature Extraction [21] is method selecting and combining variables into features, effectively reducing the volume of data to be handled while still processing the original data set accurately and completely. Extraction of features can also decrease the quantity of redundant data for a particular assessment. The tests are directed to understand the effectiveness of performance and validate the efficiency of features mined from the two class and multiclass approach based on the NSL-KDD dataset. Training (NSLKDDTrain+) and testing (NSLKDDTest+) data are used separately for training and testing, respectively.

A. Autoencoder

An autoencoder (AE) neural network is an unsupervised machine learning algorithm that uses backpropagation to set goal values equal to the inputs. They are used in a smaller representation to reduce the size of the inputs and will recreate it from the compressed data if anyone wants the original data. AE exploits a balanced structure shown in Fig. 2 that consist of an encoder that constrict input into a fewer bits that comprise the actual information and a decoder part skilled to renew the input from the encoder's extracted features, each has a neural network with multiple hidden layers that are generally positioned evenly. It holds an unseen layer which studies the latent depiction of the input vector with smaller dimensions in a different feature space. The hidden layer of autoencoder, called bottleneck has lesser nodes than the input and the output layer. Then AE is called undercomplete. This is a method for deciding which aspects of observed data are appropriate information and which aspects can be rejected. Training task in an under-complete AE allows it to capture the utmost substantial features of bottleneck layer training data in order to recreate the input at the output layer. This is achieved by minimizing the loss function L(x, g(f(x))) which penalizes the difference between g(f(x)) and x. At this time, the data output of the hidden layer units is the maximum low-dimensional representation of the original data and contains all the information in the original data. AEs are created from numerous layers that link the outputs of the previous layer to the inputs of the next layer. Autoencoders will compress data just like they were educated on. Compared to the original inputs, the decompressed outputs will be reduced. Training specialized algorithm instances that will perform well on a particular type of input is simple. Upon receipt of normal data, the AE will produce similar outputs. With abnormal data, the AE must produce substantially dissimilar outputs and can therefore distinguish the abnormal data.

TABLE. I. CATEGORY OF ATTACKS

Category	Attacks
DoS	Neptune, Smurf, Pod, Land, Back, Udpstorm, process-table, mail bomb, Teardrop, Apache.
U2R	Buffer overflow, perl, rootkit, spy, Ps, Http tunnel, sql attack, worm, snmp guess, load module, Xterm.
R2L	Guess-password, ftp-write, Multihop, Warezmaster, Warezclient, snmpgetattack, Named, Xlock, Xsnoop, Send-mail, Imap, Phf.
Probe	Port-sweep, IP-sweep, Satan, Mscan, Nmap, saint.

TABLE. II. PROTOCOL WISE DISTRIBUTION IN NSL KDD DATASET

Protocols	TCP	UDP	ICMP
Count	18880	2621	1043

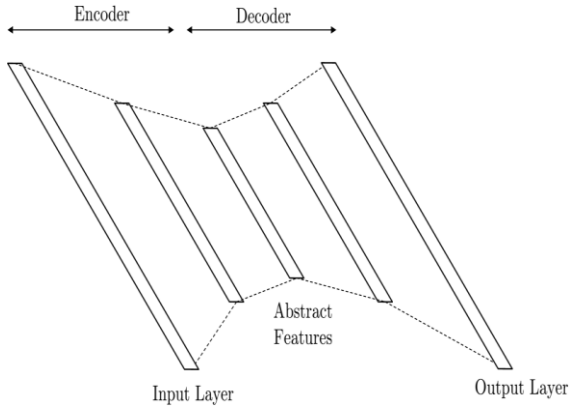

Fig. 2. Basic Structure of Autoencoder.

The method of encoding the hidden layer:

$$H = g_{\theta 1}(X) = \sigma(W_{ij}X + \phi_1) \quad (2)$$

The method of decoding the reconstruction layer from the hidden layer:

$$Y = g_{\theta 2}(H) = \sigma(W_{jk}H + \phi_2) \quad (3)$$

where $X = \{x_1, x_2, \ldots, x_n\}$ the input data vector, $Y = \{y_1, y_2, \ldots, y_n\}$ is the input data's reconstruction vector and $H = (h_1, h_2, \ldots, h_m)$ is the hidden layer's low dimensional vector output, $X \in R^n$, $Y \in R^n$ and $H \in R^m$ in which n is the input vector's dimension and m is the no. of veiled units. $W_{ij} \in R^{m \times n}$, the matrix of the weight relation between the input and the hidden layer. $W_{jk} \in R^{n \times m}$, weight matrix of the hidden layer and the reference layer. $\phi_1 \in R^{n \times 1}$ and $\phi_2 \in R^{m \times 1}$ are the input and hidden layer bias vectors respectively. $g_{\theta 1}()$ and $g_{\theta 2}()$ are stimulation feature of the hidden layer neurons and the output layer neurons correspondingly, whose functions are to map to [0,1] the network summation result.

B. Convolutional Autoencoder

CAEs are identical to AE, but the distinction is that all the input locations are shared by weights in the CAE [24], maintaining the spatial position like CNN. Convolution Neural Networks (CNNs) are for handling high-dimensional data with some spatial denotation and the data can be images, video, speech sound signals, text sequence of characters, or any other multidimensional information. The loss function is same as autoencoder as given in equation (4).

$$e(x,y,W) = \frac{1}{2N}\sum_{i=1}^{n} ||x_i - y_i||_2^2 + \lambda \|W\|_2^2 \quad (4)$$

λ is the regularization parameter for the regularization term.

CAE comprehends convolutional, deconvolutional, pooling, and unpooling layers. Convolutional layer outlines the data of a filter into a scalar with different parameters. In the function map, it joins multiple input activations in a filter's fixed receptive field to a singly activation output. Low-level features of the input frames are extracted in the initial layers of convolution layer and high-level features in later layers and vice versa in deconvolution layers. The pooling layer was planned for entirely supervised feedforward manners and shows a constant factor in the latent representation. Moreover, permits composite representations but lessens the three-dimensional size of representations by reducing the number of parameters and computation. Unpooling layer accomplishes the inverse pooling operation, reconstructing the original size of individually quadrilateral sub-region.

CAEs are state-of - the-art tools to learn convolutional filters in an unsupervised manner and then can be tested to any input to extract features. Instead, these features can be used to perform any function requiring a compact representation of the data, such as classification. CAEs [9] scale fine to realistic high-dimensional data due to their convolutionary nature, as the numeral of parameters essential to yield an activation map is permanently the same inspite of the input size. The encoder selects features over convolution and pooling layers and the decoder rebuild the input over unpooling and reordered convolution layers. Each decoder layer equivalent to that in the encoder shall be located in the reverse sequence of the encoder layers. Initially the input data is transformed to binary image using character-level binary image transformation technique. Then the output of this is fed into two convolution and deconvolution layers, two pooling layers and unpooling layers for feature extraction.

Consider the message's maximum permissible length is X and any character that exceeds it is neglected. The message is therefore converted into 68x1x X. For the given kth character within the permitted characters, all its positions are found inside the data. Then, their respective channel k locations in the picture are set to 1. The steps involved in this transformation technique are initially the given data is converted into reverse order and transform each character into a vector with a specific length. Then transform a set of vectors into one dimensional image with the specified number of channels. Image matrix is converted into an array, rescale it between 0 and 1.

The latent representation of the k-th feature map for a mono-channel input x is given by

$$h^k = \sigma(x * W^k + b^k) \quad (5)$$

Where the bias is transmitted to the entire map, σ is an activation function, ∗ signifies the 2D convolution. The minimizing cost function is the mean squared error

$$E(\theta) = 1/2n \sum_{k=1}^{n} (x_k - y_k)^2 \quad (6)$$

As the backpropagation algorithm is used for standard networks to measure the gradient of the error function with respect to the parameters. Convolution operations can effectively achieve this with the following formula.

$$\delta E(\theta)/\delta W^k = x * \delta h^k + h^k * \delta y \quad (7)$$

δh and δy are the deltas of the hidden states and the reconstruction of the hidden states, respectively. Using stochastic gradient descent, the weights are then updated.

1) Classification: The output from extraction of the CAE features was transferred to a classifier to be categorized using two separate classification, the binary classification that tells attack or normal data and the five classification that includes four class of attacks and the normal. Softmax is a soft version of max function. This divides the whole (1) instead of choosing a maximum value with the highest element having the largest portion of the distribution, but other smaller elements do get some of it [3]. This softmax property which outputs a distribution of probabilities appropriate for probabilistic clarification in classification tasks. We use this as the last layer in neural networks because of the necessary property of softmax function outputting a probability distribution. To do this, the derivative or gradient must be measured and transferred back to the preceding layer through backpropagation.

$$\delta p_i/\delta a_j = (\delta ea_i / \sum_{k=1}^{n} ea_k) / \delta a_j \quad (8)$$

Cross entropy reveals the difference between the assumption of distribution of output and the original distribution. This is considered a loss function in neural networks that have output layer softmax activations. It is defined as

$$H(y, p) = - \Sigma_i y_i \log (p_i) \quad (9)$$

Loss function tests how consistent the set of parameters in the training dataset is with respect to ground truth labels. The loss function has been established in such a way that good training data predictions are tantamount to having a small loss.

IV. Evaluation Discussion

The anticipated IDS framework is tested on the NSL-KDD dataset that consists of approximately 22,544 features and has huge quantity of network traffic information, marked as usual or abnormal. The performance assessment of collaborative unsupervised machine learning is finished using NSLKDD training and testing data. Train datasets were used to train the model of machine learning and test datasets were accustomed to evaluate the trained model of machine learning. There are four possible states for each activity observed, in terms of the performance metrics of an IDS. The measures of the assessment are considered and measured and can be described as:

True positive (TP): irregularity decently characterized as anomalousness.

False positive (FP): irregularity poorly characterized as anomalousness.

True negative (TN): regular data correctly characterized as unusual.

False negative (FN): irregularity inaccurately characterized standard.

Accuracy: say the exact classification fraction of all records in the test set as shown in (10).

Precision: say the right intrusion estimate fraction with predictable overall intrusions as in (11)

Recall: say the allowed intrusion estimate fraction separated by the full amount of valid intrusion possibilities in the test set in (12).

$$A=(TP+TN) / (TP+TN+FP+FN) \quad (10)$$

$$P=TP/(TP+FP) \quad (11)$$

$$R=TP/(TP+FN) \quad (12)$$

ROC Curves summarize the trade-off for a predictive model using different probability thresholds between the true positive rate and the false positive rate. ROC curves depend on the true positive, true negative, false positive and false negative. RoC is a plot of False Positive Rate (FPR) of binary classifiers against True Positive Rate (TPR). Area under RoC Curve (AuC) is a measure of how well a binary classifier can accomplish label predictions. This shows the performance of a classical model for a binary classifier.

V. Outcomes

Performance evaluation is done on testing data using CAE and softmax classifier. The experiments were performed on the basis of the NSL-KDD dataset to check performance efficiency and verify the reliability of the low-dimensional characteristics obtained from our two-class and multi-class classification strategy. Moreover, it compares the performance with the existing methods and several recent approaches like SVM, KNN, STL IDS, CNN.

In Fig. 3, execution metrics such as precision, recall and accuracy of CAE- Softmax is compared on training dataset. Accuracy, Precision and recall for two class training data are 99.9, 99.5, 99.5 respectively and five class training dataset are 97.92, 99.39, 99 respectively. Assessment of the same on test data for accuracy, precision and recall is depicted in Fig. 4 with values 92,91,91.05 for two class categories respectively and 97,95,91 for five class categories respectively. So, after several models have been introduced and evaluated, results show that the CAE- Softmax model being proposed has better performance. The planned method is then analyzed to the present algorithms as in Fig. 5 to give a better accuracy. The graph outcome of types of protocols is shown in Fig. 6. Fig. 7 shows the ROC Curve for NSLKDD dataset. These show that the model reduces the false alarm levels to an acceptable level to retain total safety against serious attacks. This system provides high detection rate.

Table III shows the comparison of precision, recall, accuracy the projected model on training data for two class and multiclass. Table IV illustrates the same evaluation method of the model on the test data. The accuracy of the existing IDS algorithms with the CAE – Softmax IDS is compared and the values are given in Table V. The experimentation outcome shows the attacks in NSL KDD test data as in Table VI.

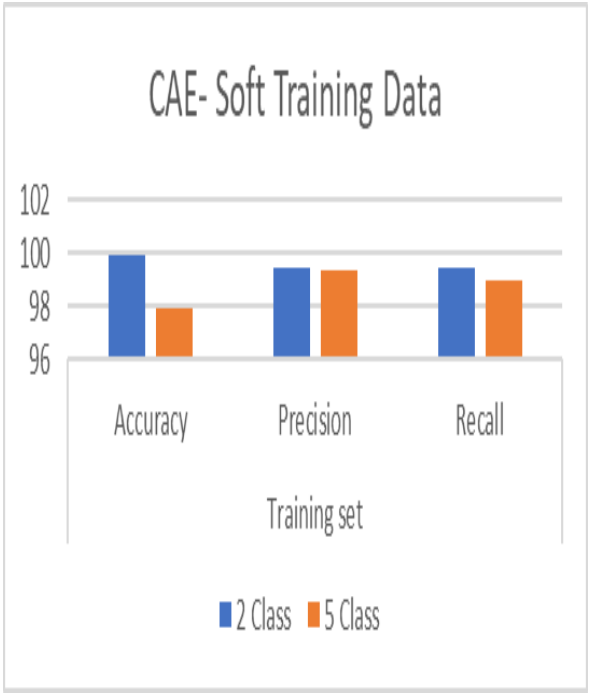

Fig. 3. Assessment of CAE-Soft IDS on Training Data.

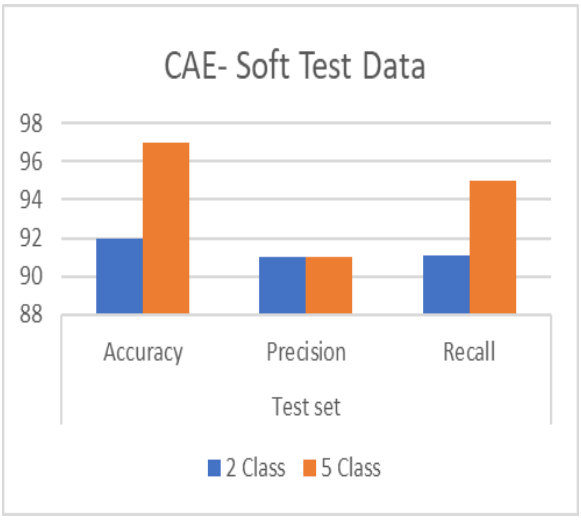

Fig. 4. Assessment of CAE-Soft IDS on Test Data.

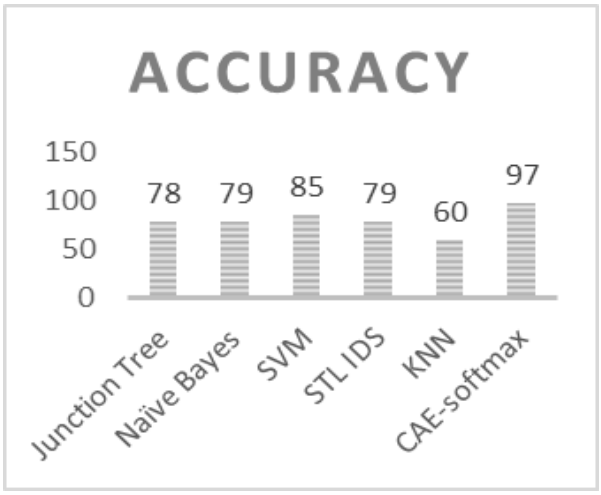

Fig. 5. Existing IDS Versus CAE-Softmax.

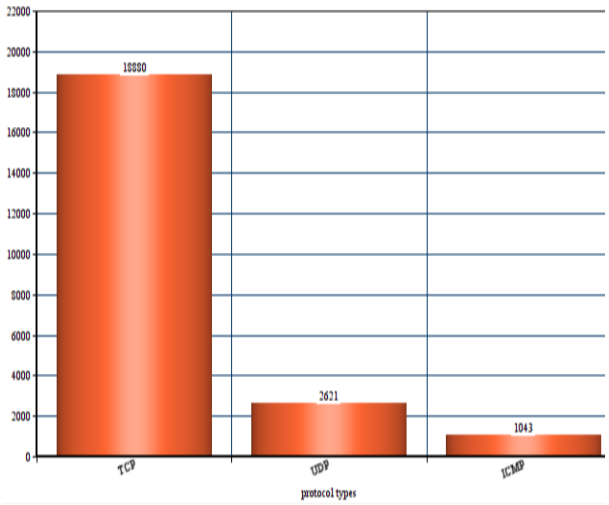

Fig. 6. Protocol Types in NSLKDD.

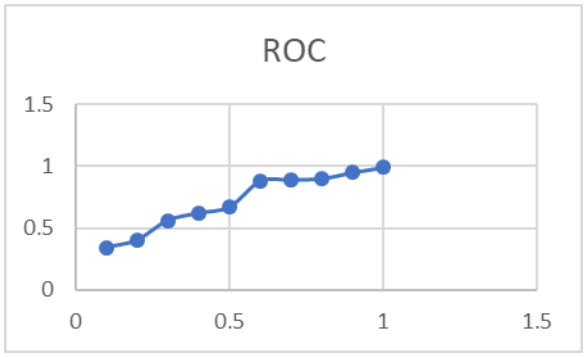

Fig. 7. ROC Cirve for NSLKDD Data.

TABLE. III. PERFORMANCE METRICS ON TRAINING DATA

	Training set		
	Accuracy	***Precision***	***Recall***
2 Class	99.9	99.5	99.5
5 Class	97.92	99.39	99

TABLE. IV. PERFORMANCE METRICS ON TEST DATA

	Test set		
	Accuracy	***Precision***	***Recall***
2 Class	92	91	91.05
5 Class	97	91	95

TABLE. V. COMPARISON OF ACCURACY OF ALGORITHMS

ALGORITHM	**ACCURACY**
Junction Tree	78
Naïve Bayes	79
SVM	85
STL IDS	79
KNN	60
CAE-softmax	97

TABLE. VI. COUNT OF ATTACKS IN NSL KDD

Attacks	**Count**
Normal	9711
DOS	7456
Probe	2421
U2R	200
R2L	2756

VI. CONCLUSION

The suggested CAE-Softmax IDS scheme is an enhanced method of intrusion that utilizes methods of machine learning to select and classify features. This technique is a pledge to reduce false positive as well as false negative. The above model analyzed the convolutional autoencoder, Softmax Classifier and existing IDS SVM, KNN, STL methods and outperformed present diverse methods in testing precision and training. By applying this to the actual network to implement it more effectively, further step can be taken. This can be applied to an improved efficiency for all class categories.

Furthermore, IDS outputs can be presented to any real time applications like investigations to construct timeline of an attack and associate attacks to find out the trespasser.

REFERENCES

[1] R. Vinayakumar, Mamoun Alazab, K. P. Somani, Prabaharan Poornachandran, Ameer Al-Nemrat, Sitalakshmi Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System" IEEE Access Volume 7, 2019 page. 41525 -41550.

[2] Dhikhi T, M. S. Saravanan "An Enhanced Intelligent Intrusion Detection System using Machine Learning" International Journal of Innovative Technology and Exploring Engineering, Vol 8, Issue 9, July 2019, pp. 2177-2181.

[3] Xin Ye And Qiuyu Zhu 'Class-Incremental Learning Based on Feature Extraction of CNN With Optimized Softmax and One-Class Classifiers' IEEE Access, Vol 7, 2019 pp. 42024-42031.

[4] Haipeng Yao, Danyang Fu, Peiying Zhang, Maozhen Li, and Yunjie Liu MSML: A Novel Multilevel Semi-Supervised Machine Learning Framework for Intrusion Detection System IEEE Internet Of Things Journal, Vol. 6, No. 2, April 2019.

[5] Gael Kamdem, Momo ZIAZET 'Convolutional Neural Network for Intrusion Detection System In Cyber Physical Systems', ResearchGate, May 2019.

[6] Nathan Shone, Tran Nguyen Ngoc, Vu DinhPhai, Qi Shi, "A Deep Learning Approach to Network Intrusion Detection", IEEE Transactions on Emerging Topics in Computational Intelligence, Vol. 2, No. 1, February 2018, pp. 41-50.

[7] MajdLatah, LeventToker, "Towards an efficient anomaly-based intrusion detection for software-defined networks" IET Netw., 2018, Vol. 7 Iss. 6, pp. 453-459.

[8] Majjed Al-Qatf, Yu Lasheng, Mohammed Al-Habib, And Kamal Al-9Sabahi "Deep Learning Approach Combining Sparse Autoencoder With SVM for Network Intrusion Detection" IEEE. Translations and content mining, VOLUME 6, 2018, pp. 52843-52856.

[9] Marco Maggipinto, Chiara Masiero,Alessandro Beghi,Gian AntonioSusto 'A Convolutional Autoencoder Approach for Feature Extraction in Virtual Metrology' , Procedia Manufacturing, Volume 17, 2018, Pages 126-133.

[10] Iftikhar Ahmad, Mohammad Basheri, Muhammad Javed Iqbal, And Aneel Rahim" Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection" IEEE Transactions on Special section on survivability Strategies for Emerging Wireless Networks, Volume 6 May 2018 pp. 33789-33795.

[11] Congyuan Xu, Jizhong Shen, Xin Du, and Fan Zhang "An Intrusion Detection System Using a Deep Neural Network With Gated Recurrent Units" IEEE,Volume: 6, 2018, Page(s): 48697 – 48707.

[12] Sheraz Naseer, Yasir Saleem, Shehzad Khalid, Muhammad Khawar Bashir, Jihun Han, Muhammad Munwar Iqbal, And Kijun Han" Enhanced Network Anomaly Detection Based on Deep Neural Networks" IEEE Transactions on Special Section on Cyber-Threats and Countermeasures in The Healthcare Sector Volume 6, 2018 pp.48111-48246.

[13] Mohammed HasanAli, Bahaa Abbas Dawood Al Mohammed, Alyani Ismail, and MohamadFadliZolkipli "A New Intrusion Detection System Based on Fast Learning Network and Particle Swarm Optimization" IEEE Transactions, Volume 6, 2018, pp. 20255-20261.

[14] PeiyingTao, Zhe Sun, And Zhixin Sun "An Improved Intrusion Detection Algorithm Based on GA and SVM" IEEE Transactions on Special Section on Human-Centered Smart Systems and Technologies, Volume 6,2018 pp. 13624-13631.

[15] Qiangyi Zhang, YanpengQu, Ansheng Deng "Network Intrusion Detection Using Kernel-based Fuzzy-rough Feature Selection", IEEE International Conference on Fuzzy Systems,2018.

[16] HuijunPeng, Chun Ying, Shuhua Tan, Bing Hu, And ZhixinSun, "An Improved Feature Selection Algorithm Based on Ant Colony Optimization", IEEE Transactions Volume 6, 2018, pp. 69203-69209.

[17] Antonia Nisioti, AlexiosMylonas, Paul D. Yoo, and VasiliosKatos "From Intrusion Detection to Attacker Attribution: A Comprehensive Survey of Unsupervised Methods", IEEE Communications Surveys & Tutorials, VOL. 20, NO. 4,2018, pp. 3369-3388.

[18] Zheng Wang, "Deep Learning-Based Intrusion Detection with Adversaries", IEEE Transactions on Special Section on Challenges and Opportunities of Big Data Against Cyber Crime, Volume 6, 2018, pp.38367-38384.

[19] L. M. Ibrahim, D. T. Basheer, and M. S. Mahmod, ''A comparison study for intrusion database (KDD99, NSL-KDD) based on self organization map (SOM) artificial neural network,'' J. Eng. Sci. Technol., vol. 8, no. 1, pp. 107–119, 2013.

[20] F. A. B. H. Ali and Y. Y. Len, ''Development of host based intrusion detection system for log files,'' in Proc. IEEE Symp. Bus., Eng. Ind. Appl. (ISBEIA), pp. 281–285, Sep. 2011.

[21] S. Rifai, P. Vincent, X. Müller, X. Glorot, and Y. Bengio, ''Contractive auto-encoders: Explicit invariance during feature extraction,'' in Proc. 28th Int. Conf. Int. Conf. Mach. Learn. (ICML), 2011, pp. 833–840.

[22] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, ''A detailed analysis of the KDD CUP 99 data set,'' in Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl. (CISDA). Piscataway, NJ, USA: IEEE Press, 2009, pp. 53–58.

[23] C.-F. Tsai, Y.-F. Hsu, C.-Y. Lin, and W.-Y. Lin, ''Intrusion detection by machine learning: A review,'' Expert Syst. Appl., vol. 36, no. 10, pp. 11994–12000, 2009.

[24] Seungyoung Park, Myungjin Kim, Seokwoo Lee 'Anomaly Detection for HTTP Using Convolutional Autoencoders' IEEE Access, Volume: 6, Page(s): 70884 - 70901.