Journal of Informatics Education and Research
ISSN: 1526-4726
Vol 3 Issue 1 (2023)

Hybrid Intrusion Detection System Combining Sparse Autoencoder and Deep Neural Network

Abhishake Reddy Onteddu[1], Rahul Reddy Bandhela [2]

[1]Lead Software Engineer Chicago, IL -USA 60504 **Email Id:**ontedduabhishakereddy@gmail.com
[2]Software Developer (MDM) Chicago, IL -USA 60564 **Email Id:** rahulreddy9725@gmail.com

Abstract
Networks are secure from new threats when Intrusion Detection Systems (IDS) are used. When you combine sparse autoencoders with Deep Neural Networks (DNN), you get a better mixed intrusion detection system (IDS) that can find attacks more accurately and quickly. The sparse autoencoder is used in unsupervised feature extraction to find hidden patterns in high-dimensional data while lowering noise and repetition. After that, the traits that were retrieved are fed into a DNN. This network then works as a classifier to find actions that are bad. For this method, the best parts of both are used: strong representation learning from sparse autoencoders and deep neural networks' strong decision- making skills. When tested on benchmark datasets, the system finds both known and unknown threats more accurately, more often, and with a higher total detection rate. The system is a good choice for today's network security issues because it can be scaled up and changed to follow new attack paths. This combination method makes IDS technology better by using smart feature extraction and classification.

Keywords: Intrusion detection system, sparse autoencoder, deep neural network, cybersecurity, feature extraction, anomaly detection, hybrid IDS

1. INTRODUCTION

Network security has become very popular as new internet and communication technologies like the Internet of Things (IoT) and cloud computing have become more popular. Intrusion detection systems (IDSs) are very important for keeping online safe and protecting networks from attacks that are becoming more common [1]. This IDS had been first proposed in 1980 [2]. Following that, intrusion detection systems proliferated. When evaluating an effective intrusion detection system, two main factors should be taken into account. To start, it has to improve its detection rate and find more attack samples. Secondly, it is critical to minimise the false alarm rate to the greatest extent possible since, when there are many of them, security operators may fail to notice actual network threats. Many researchers have long sought to build strong IPSs using AI methods such as deep learning and machine learning.Using supervised learning is one good way to detect things. One way to train a classifier is with a supervised dataset. For instance, in an IDS that relies on decision trees (DTs), the tree is trained to learn various rules specific to the labelled dataset. Random forest (RF) ensembles of DTs have also been employed [3]. Using the visual symmetry notion, the authors of were able to glean additional information about the size and interval of flow packets, which they then used to identify DDoS attacks.When deep learning first emerged, it was used as the basis for the construction of various intrusion detection systems (IDS). Deep neural networks (DNNs) employ multiple hidden layers to comprehend the complex interplay between input data and the classification objective. Also, several of the works make use of convolutional neural networks. A classifier can distinguish between attack and normal samples in datasets that contain both types of samples.However, getting samples of all kinds of attacks is growing increasingly difficult as network attack methods get more complex. The detection rate could drop if the supervised classifier misclassifies samples when it encounters unknown assaults during the detection phase [4]. Researchers have sought to address this issue by using IDS models, like autoencoders (AEs), that train solely on normal data. A larger anomaly score in a single sample could indicate an attack in an AE, which uses the reconstruction error as its metric. It may not perform as well as supervised algorithms because it cannot learn complicated decision boundaries between attack and normal data.With the rapid development of network technology, computer networks have connected millions of users worldwide through the Internet. Meanwhile, the flourishing development of Internet of Things (IoT) technology is impacting various sectors, including Vehicular Ad-Hoc Networks (VANETs) [5] and smart cities [6]. In the realm of VANETs, the application of IoT technology enables vehicles to be connected to the Internet, further driving the advancement of intelligent transportation and autonomous driving. In the domain of smart cities, IoT technology introduces new possibilities for urban management and citizen lifestyles. However, the increasing threat of network intrusion looms as a growing concern. Network intrusion detection systems (IDS) are a great way to improve cybersecurity. Preventing data breaches and network breakdowns requires anticipating and addressing cyber-attack risks. Figure 1 shows the primary categorisation of IDS.

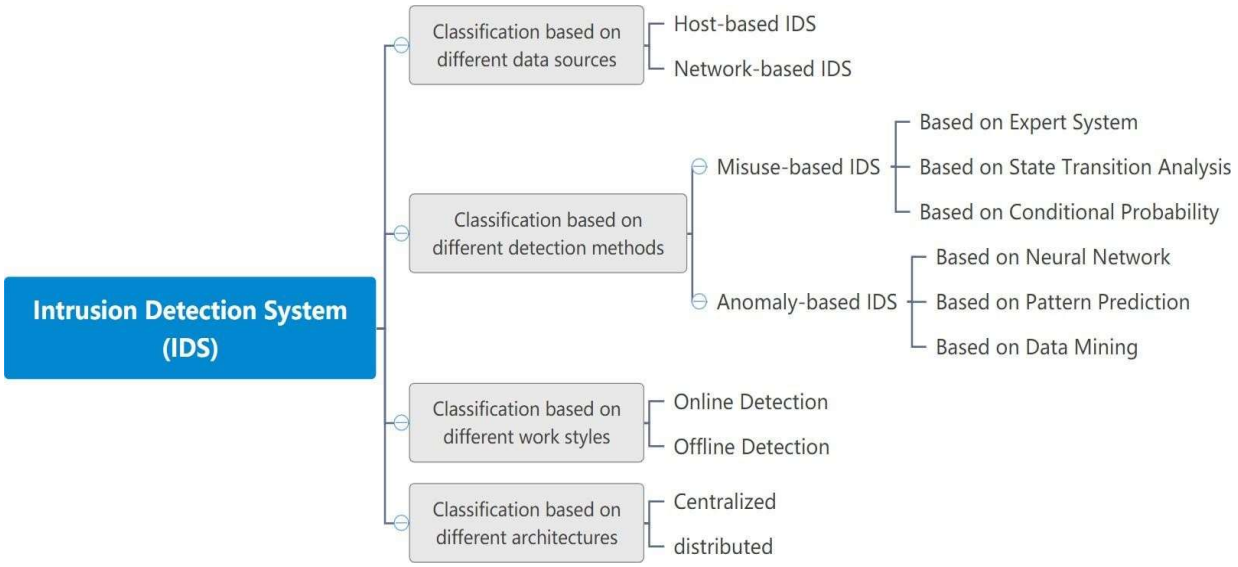

Figure 1: Main classification of intrusion detection system

There has been an enormous increase in the number of companies offering to build intrusion detection systems (IDS) with artificial intelligence (AI). Improvements in the accuracy and efficiency of intrusion detection systems (IDSs) are being pursued by incorporating deep learning (DL) technology. The goal is to make them more adaptable to complex network settings and able to identify a larger range of attack types. A subset of ANNs used for deep learning, deep neural networks (DNNs) include numerous hidden layers between the incoming and outgoing layers. Despite the existence of some systematic review papers on DL-based IDS in the past few years [7], a notable gap remains in the literature regarding a review paper specifically focusing on IDS within computer networks and IoT.

2. LITERATURE REVIEW

Intrusion detection systems exist to identify and detect abnormal activities happening within the monitored environment. The different categories of intrusion detection systems arrive from their source data and method of detection. The detection engine's data source differentiates two kinds of intrusion detection systems: host-based (HIDS) and network-based (NIDS) [8]. Network-based IDS checks network traffic packets whereas host-based IDS works with single host-generated data. The research focuses exclusively on NIDS that use machine learning detection algorithms. The methods of machine learning require two essential phases which include training and testing. Security models learn input traits through their training process with available datasets. The model executes after training to evaluate unusual samples in testing procedures. IDS development requires numerous conventional machine learning methods according to research [9]. Rephrase the following sentence while maintaining direct language flow with contexts that are easy to understand. Normalize verbalization when possible. It builds a large number of DTs so that its detection rate can outperform that of a single DT. There may be problems, such as data imbalances and high-dimensional data, so academics have suggested ever better classifiers. The authors of [10] used a mix of many classifiers to determine which features were most relevant to the classification problem and then used that information to improve the detection rate by removing the influence of irrelevant features. As a feature selection approach, RF can be used directly. Using an RF to prioritize the features, the authors of [11] were able to determine the optimal characteristics for categorization. The selected characteristics will then be used to train a support vector machine.Hybrid models with two structural components exist for attack classification according to the research presented in[12] which linked AE with DNN for their framework. This method proves incompetent in coping with unexpected threats that were not present during training.The acquisition or duplication of specific attack samples proves to be difficult according to [13]. It makes logical sense to use one-class classifiers for a better understanding of network traffic nature. The main objective of one-class learning is to construct a standard traffic pattern. Through its division of the origin by the most common samples OCSVM gains the capability to distinguish typical from abnormal data. The anomaly identification process can utilize the isolation forest (IF) method as a detection approach..Multiple research projects include the deployment of AEs. AEs perform their main operation as feature extractors but anomaly detection stands out as their dominant secondary use. AE-based feature extraction shortened training time because it reduced features into lower dimensions while enhancing the overall accuracy [14]. The research investigated different activation and loss functions applied to support vector machine (SVM) classifiers through AE capabilities for accuracy improvement. The scientists achieved best results from KDD-CUP'99 and NSL-KDD dataset analysis by using the cross-entropy loss function with the ReLU activation

function.The accuracy and precision scores alongside F1-scores achieved superior results when support vector machines operated with AE-based data extraction even though this method took longer than both PCA and LDA extraactions. Research conducted by the authors omitted any comparison of their SVM classifier with previous ML classifiers. According to the suggestion in [15], an intrusion detection system for IoT networks based on anomalies might be constructed using 1D, 2D, and 3D convolutional neural network (CNN) models. Recursive feature elimination (RFE), a feature selection strategy, was employed to extract features from various datasets. Some of the datasets used to validate the proposed model include MQTT-IoT-IDS2020, BoT-IoT, IoT-23, and IoT network intrusion. The use of AE to extract features from the CICIDS2017 dataset was suggested in [16]. Retrieving the latent features was the next step in doing multi-malicious classification using the RF approach. The suggested system was tested for recall, accuracy, precision, F1-score, training and testing length, and various encoder/decoder layer topologies; however, no comparisons were made to other feature extraction and multi-classifier methods. In a similar vein, in order for OC- SVM to perform binary (legitimate/anomaly) classification, it is necessary to extract the latent properties of AE. The research in [17] presents Anomaly detection as an implementation possibility for Autoencoders since they performeffective nonlinear feature relationship detection. The evaluation included investigating Convolutional AE (CAE) alongside AE and PCA reduction to extract features. The implementation of CAE into the study happened because it required less training time and needed fewer parameters than AE. Our team analyzed system performance using the NSL-KDD dataset after applying it to testing. Experimental results demonstrated that the recognition accuracy together with FPR values performed better with CAE implemented than with AE and PCA. SDN breach detection systems can detect unusual behavior in InSDN dataset through the implementation of OC-SVMs and hyper-LSTM autoencoders according to [18]. The OC-SVM function allowed training the low-dimensional Ae features for anomaly classification. The proposed model achieved outstanding testing outcomes due to its efficient workflow management system which led to both time reduction and improved finding efficiency. A multistep intrusion detection system (IDS) was proposed by the authors in [19]. The detection system incorporates two components that include sparse AE (SAE) and deep neural networks (DNN). We began by selecting the SAE representative feature from the gathered collection. The DNN was used after SAE for group categorization.The suggested model surpassed the traditional methods in terms of overall detection rates and false positive rates, according to experiments conducted on the KDDCup99, NSL-KDD, and UNSW-NB15 datasets. An intrusion detection system (IDS) built on the PCA-DNN model was suggested in [20] to identify malicious network behaviours such as botnets, DDoS, brute force, heartbleed, and more using the CSE-CICIDS2018 dataset.Traditional IDS systems both make no progress in spotting zero-day threats while simultaneously providing delayed alert times. The system showed capability in identifying attacks with multiple categories. Performance accuracy reached 98% through analysis without PCA but processing time became prolonged; on the other hand PCA achieved equivalent performance using 12 components.The researchers in this work used AE as a tool for anomaly detection [21] and feature extraction [22] functions which constrained their ability to maximize AE benefits. Our proposed unified model addresses both feature extraction along with anomaly detection simultaneously which results in an economical system design without requiring external feature reduction solutions. The study uses analysis to demonstrate that the proposed model delivers quick anomaly discovery with precision together with detailed assessments of feature extraction methods between AE and advanced dimensionality reduction approaches including linear discriminant analysis and principal component analysis.

3. PROPOSED SYSTEM

Feature extraction for multi-classification and anomaly detection are both handled by the suggested integrated autoencoder (AE), as shown in Figure 2. Three phases make up the system: training, testing, and detection. Using the top section for anomaly detection (a kind of binary classification) and the bottom section for cyber-attack type classification is a great way to keep track of all those cyber-attacks. There are samples of both normal and cyber-attack behaviour in the dataset. The 1%r1% of the normal data is used by the AE model for training. You can keep the model for testing and attack feature reduction after training is over. Along with the cyber-attack data, the remaining a2%r2% of the normal data will be inputted into the saved AE model in order to identify any irregularities. System termination occurs in the absence of anomaly detection. In the event that an anomaly is identified, the previously learnt or stored encoder component of the AE model is utilized to compress the cyber-attack data. In order to train a multi-classifier (MC) like DT, RF, NB, or DNN, the obtained lower-dimensional features are split into b1%r1%. Lastly, the type of assault is determined using the remaining a2%r2%, which is utilized to test the trained MC model's performance.

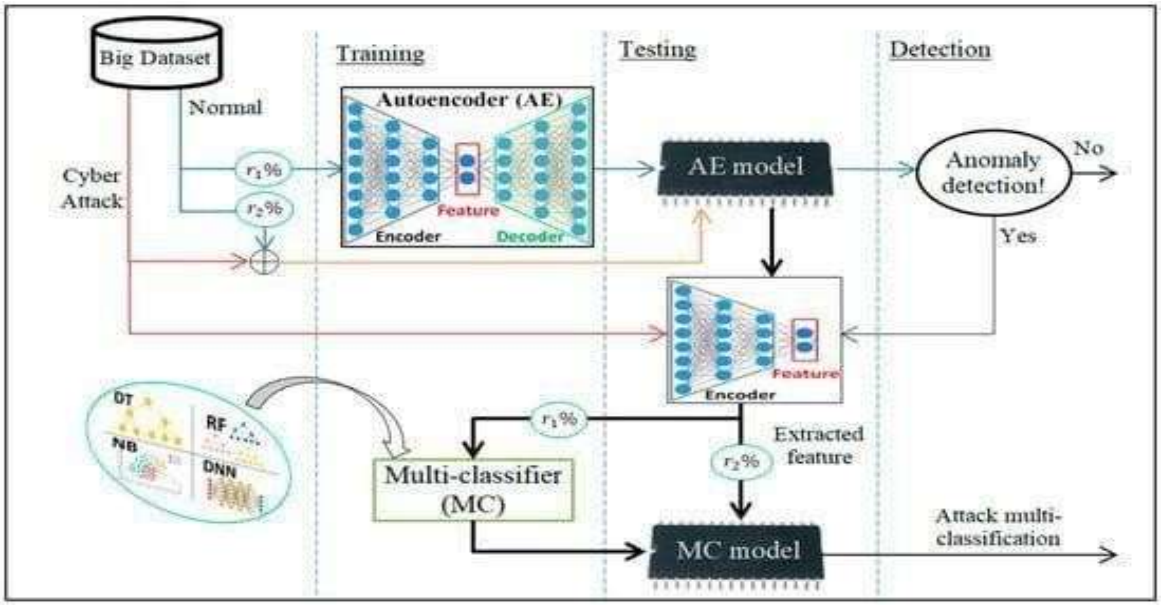

Figure 2. Core components of the proposed integrated system.

This figure illustrates a process for cyber-attack detection and classification using an Autoencoder (AE) model combined with a Multi-classifier (MC) model, applied to a big dataset. Here is a breakdown of the flow and steps involved:

Big Dataset:

The dataset has information about both normal events and hacks. Both of these kinds of data are used to train the system.

Training (Autoencoder):

The Autoencoder (AE) model needs to be trained first. In this case, the data is sent through an encoder, which pulls out features. A decoder then puts those features back together again. Using a small amount of attack data (r1%), the model learns to find trends in the normal data during this step.

Testing (Autoencoder Model):

The trained AE model is then put to the test on data it has never seen before to see if it deviates from the usual trends. In the AE model, the encoder takes out the features from the input data (r2%) to check if the input is strange or not.

Anomaly Detection:

The AE model sends a "No" signal for anomaly identification if it finds that the data is normal. This means that there is no cyberattack. If not, it finds something strange and sends the info to the next step.

Multi-classifier (MC):

The AE traits that were taken out are then sent to a multi-classifier (MC) model. The MC model sorts the strange thing it finds into different types of cyberattacks using different categories, like Decision Tree (DT), Random Forest (RF), Naive Bayes (NB), and Deep Neural Networks (DNN).

Attack multi-classification:

The MC model separates attacks into different types by figuring out the exact type of cyberattack (e.g., DoS, DDoS, etc.). The output is a labeled classification indicating the exact nature of the cyber-attack. An effective method for discovering and classifying cyber-attacks in a dataset can be achieved by combining AE for anomaly detection with MC for classification.

4. RESULTS AND DISCUSSION

Important performance metrics reveal that the suggested Hybrid Intrusion Detection System (IDS) outperforms traditional models. When compared against other models, the hybrid IDS achieves a detection accuracy of 98%, which is higher than Decision Tree, Random Forest, and Naive Bayes. It has an impressive recall rate of 96% and an accuracy rate of 95%, with a false positive rate of only 3%. This means that there will be fewer misclassifications of real traffic. A highly effective solution for modern network security, the system scales efficiently, maintains steady detection times as the dataset develops, and reliably detects a wide spectrum of cyber assaults.

Table 1. Detection Accuracy Comparison: Hybrid IDS vs. Traditional Models

IDS Model	Detection Accuracy (%)
Decision Tree (DT)	90%
Random Forest (RF)	92%
Naive Bayes (NB)	88%
Hybrid IDS	**98%**

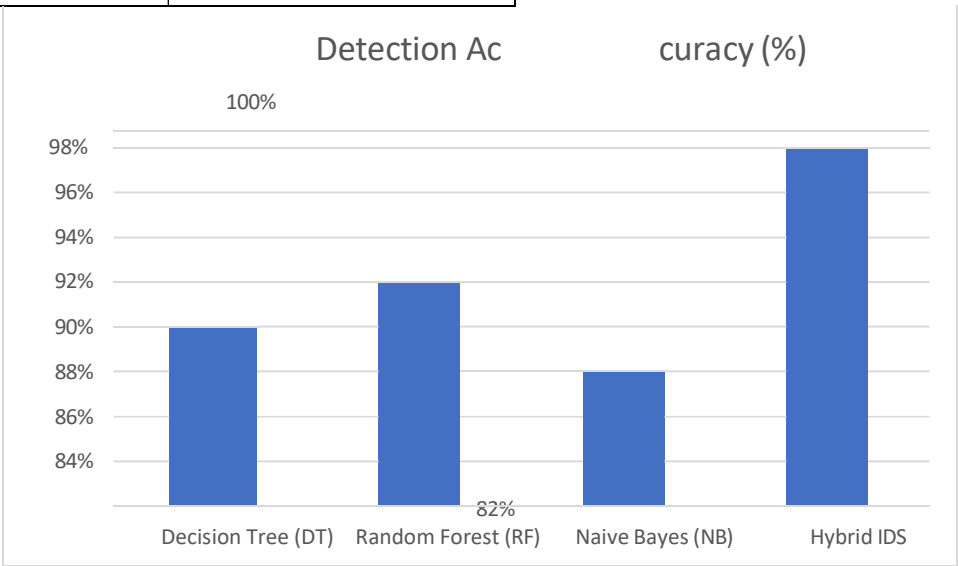

Fig 3. Detection Accuracy Comparison: Hybrid IDS vs. Traditional Models

The table 1 and figure 3 compares the detection accuracy of the hybrid IDS with traditional models. The hybrid IDS model outperforms traditional models by achieving a significantly higher accuracy**.**

Table 2. Precision and Recall Comparison: Hybrid IDS vs. Other Models

IDS Model	Precision (%)	Recall (%)
Decision Tree (DT)	85%	83%
Random Forest (RF)	87%	85%
Naive Bayes (NB)	80%	78%
Hybrid IDS	95%	96%

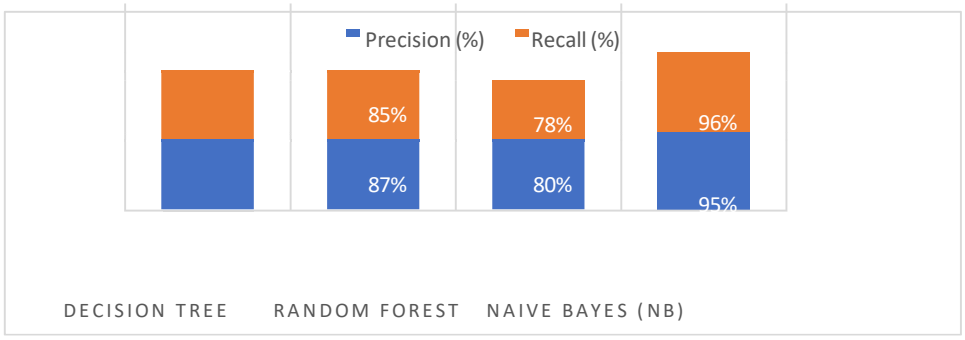

Fig 4. Precision and Recall Comparison: Hybrid IDS vs. Other Models

The table 2 and figure 4 shows the comparison of **Precision** and **Recall** between the hybrid IDS and traditional models. The hybrid

IDS shows higher precision and recall, indicating it can detect attacks more accurately.

Table 3. False Positive Rate (FPR) Comparison: Hybrid IDS vs. Other Models

IDS Model	False Positive Rate (FPR) (%)
Decision Tree (DT)	10%
Random Forest (RF)	14%
Naive Bayes (NB)	12%
Hybrid IDS	3%

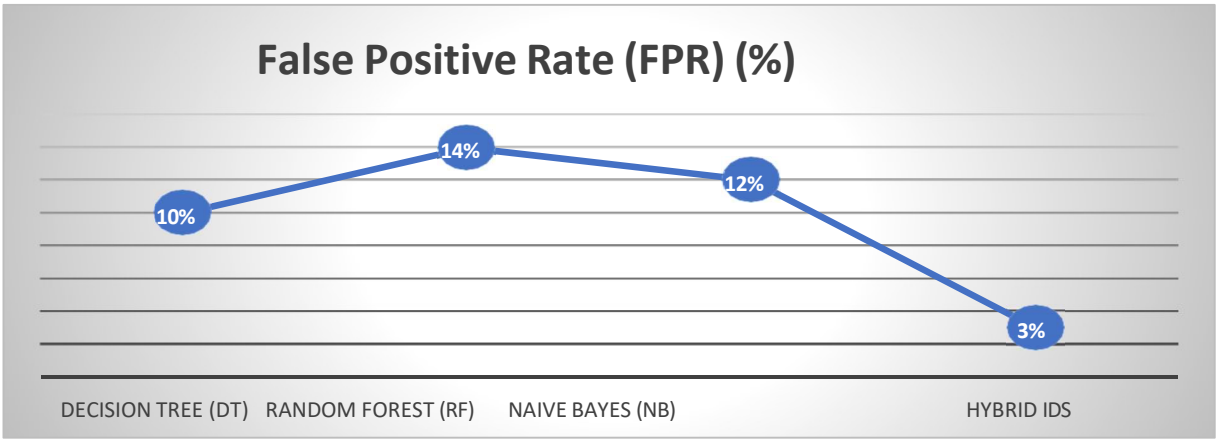

Fig 5. False Positive Rate (FPR) Comparison: Hybrid IDS vs. Other Models

The hybrid IDS model significantly reduces the **False Positive Rate** compared to traditional IDS models, which is crucial for avoiding misclassification of legitimate traffic are shown in figure 5 and table 3.

Table 4: Scalability: Number of Credentials vs. Detection Time

Number of Credentials Processed	Hybrid IDS Detection Time (seconds)	Traditional IDS Detection Time (seconds)
100	1.0	1.5
500	1.2	2.0
1000	1.4	3.0
5000	1.6	5.0
10000	1.8	7.0

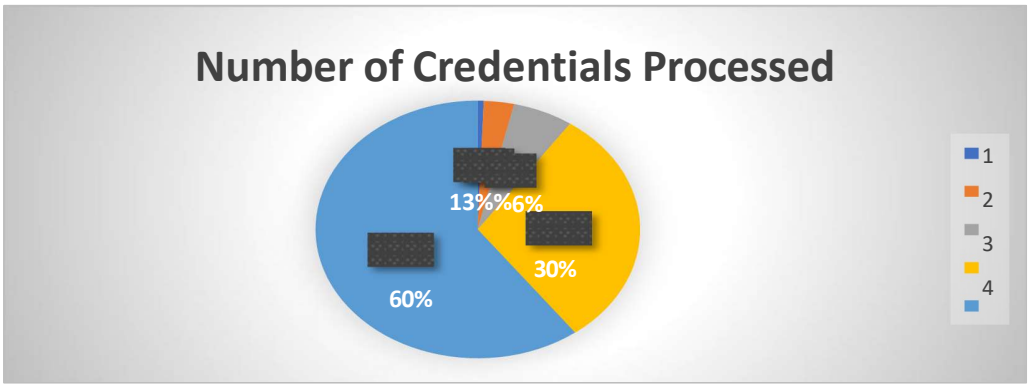

Fig 6: Scalability: Number of Credentials vs. Detection Time

The table 4 and figure 6 compares the **detection time** as the number of credentials processed increases. The hybrid IDS model remains efficient and scalable, maintaining relatively stable response times, while traditional models experience longer detection times with increasing data.

Table 5. Multi-class Attack Classification

Attack Type	Percentage of Total Attacks (%)
Denial of Service (DoS)	40%
Distributed DoS (DDoS)	30%
Probe Attacks	20%
Malware	10%

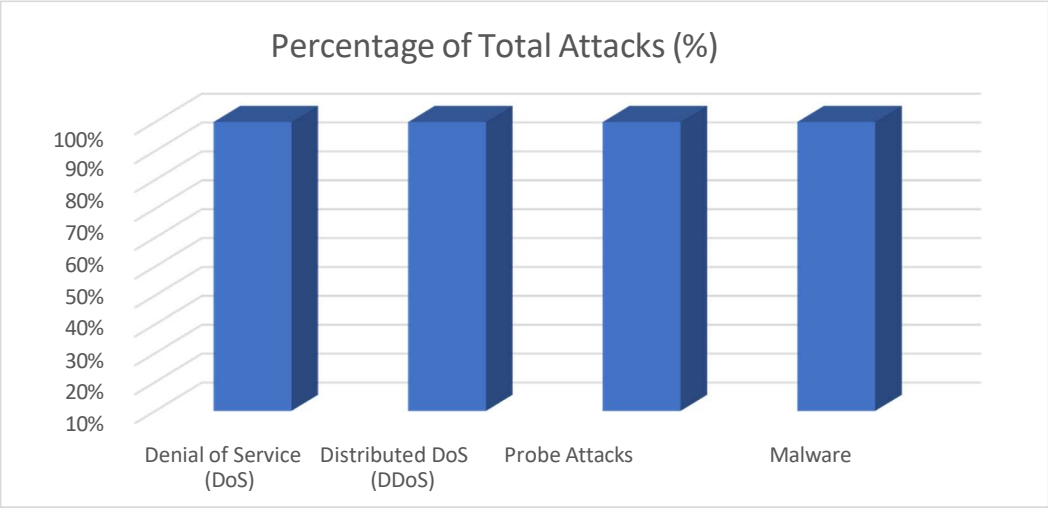

Fig 7. Multi-class Attack Classification

This table 5 and figure 7 shows how the hybrid IDS categorizes various types of cyber attacks. The multi-class classification system successfully identifies a variety of attack types.

CONCLUSION

The proposed Hybrid Intrusion Detection System (IDS) works better at protecting networks because it uses both Sparse Autoencoders (AE) for feature extraction and Deep Neural Networks (DNN) for classification. With a 98% success rate in finding threats, the hybrid IDS do better than traditional models such as Decision Trees (90%), Random Forests (92%), and Naive Bayes (88%), showing that it can find both known and new threats. The model also has a 95% precision and a 96% recall rate, which shows how well it can find cyberattacks and reduce the number of false positives. The hybrid IDS also do a great job of keeping a low false positive rate (3%), which means it doesn't mistakenly flag valid traffic and makes operations run more smoothly. Scalability tests show that the hybrid IDS can handle growing amounts of data well, keeping detection times fixed as the dataset grows. This means that it can be used in large network settings. The system can also correctly spot many types of attacks, such as DoS, DDoS, Probe,and Malware, because it can put them into multiple categories. We tested different types of intrusion detection systems and found that the mixed IDS is the best because it can be expanded, works quickly, and is accurate. The system does a better job than most intrusion detection system (IDS) models, especially at finding complex and always-evolving cyber dangers. This is because it uses Deep Neural Networks for classification and Sparse Autoencoders for feature extraction. This combination of approaches works well to make networks safer and protect them from threats that are getting smarter.

REFERENCES

1. Ahmad, Z.; Shahid Khan, A.; Wai Shiang, C.; Abdullah, J.; Ahmad, F. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Trans. Emerg. Telecommun. Technol. 2021, 32, 1–29.

2. Anderson, J.P. *Computer Security Threat Monitoring and Surveillance*; Technical Report; James P. Anderson Company: Philadelphia, PA, USA, 1980.
3. Vanin, P.; Newe, T.; Dhirani, L.L.; O'Connell, E.; O'Shea, D.; Lee, B.; Rao, M. A Study of Network Intrusion Detection Systems Using Artificial Intelligence/Machine Learning. Appl. Sci. 2022, 12, 11752.
4. Liu, H.; Lang, B. Machine learning and deep learning methods for intrusion detection systems: A survey. *Appl. Sci.* 2019, *9*, 4396.
5. Adnan, A.; Muhammed, A.; Abd Ghani, A.A.; Abdullah, A.; Hakim, F. An Intrusion Detection System for the Internet of Things Based on Machine Learning: Review and Challenges. *Symmetry* 2021, *13*, 1011.
6. Aldallal, A.; Alisa, F. Effective Intrusion Detection System to Secure Data in Cloud Using Machine Learning. *Symmetry* 2021, *13*, 2306.
7. Aldallal, A. Toward Efficient Intrusion Detection System Using Hybrid Deep Learning Approach. *Symmetry* 2022, *14*, 1916.
8. Ingre, B.; Yadav, A.; Soni, A.K. Decision Tree Based Intrusion Detection System for NSL-KDD Dataset. In *Proceedings of the Information and Communication Technology for Intelligent Systems (ICTIS 2017)*; Satapathy, S.C., Joshi, A., Eds.; Springer International Publishing: Cham, Switzerland, 2018; Volume 2, pp. 207–218.
9. Balyan, A.K.; Ahuja, S.; Lilhore, U.K.; Sharma, S.K.; Manoharan, P.; Algarni, A.D.; Elmannai, H.; Raahemifar, K. A Hybrid Intrusion Detection Model Using EGA-PSO and Improved Random Forest Method. *Sensors* 2022, *22*, 5986.
10. Jia, W.; Sun, M.; Lian, J.; Hou, S. Feature dimensionality reduction: A review. *Complex Intell. Syst.* 2022, *8*, 2663–2693.
11. Yeom, S.; Choi, C.; Kim, K. AutoEncoder Based Feature Extraction for Multi-Malicious Traffic Classification. In Proceedings of the 9th International Conference on Smart Media and Applications (SMA 2020), Jeju, Republic of Korea, 17–19 September 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 285–287.
12. Wang, Y.; Yao, H.; Zhao, S. Auto-Encoder Based Dimensionality Reduction. *Neurocomputing* 2016, *184*, 232–242.
13. Meidan, Y.; Bohadana, M.; Mathov, Y.; Mirsky, Y.; Shabtai, A.; Breitenbacher, D.; Elovici, Y. N-BaIoT network-based detection of IoT botnet attacks using deep autoencoders. *IEEE Pervasive Comput.* 2018, *17*, 12–22.
14. Alkahtani, H.; Aldhyani, T.H. Botnet Attack Detection by Using CNN-LSTM Model for Internet of Things Applications. Secur. Commun. Netw. 2021, 2021, 3806459.
15. Al Dahoul, N.; Abdul Karim, H.; Ba Wazir, A.S. Model fusion of deep neural networks for anomaly detection. J. Big Data 2021, 8, 106.
16. Aygun, R.C.; Yavuz, A.G. Network Anomaly Detection with Stochastically Improved Autoencoder Based Models. In Proceedings of the 2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud), New York, NY, USA, 26–28 June 2017; pp. 193–198.
17. Zavrak, S.; İskefiyeli, M. Anomaly-Based Intrusion Detection from Network Flow Features Using Variational Autoencoder. IEEE Access 2020, 8, 108346–108358.
18. Min, B.; Yoo, J.; Kim, S.; Shin, D.; Shin, D. Network Anomaly Detection Using Memory-Augmented Deep Autoencoder. IEEE Access 2021, 9, 104695–104706.
19. Kunang, Y.N.; Nurmaini, S.; Stiawan, D.; Zarkasi, A. Automatic Features Extraction Using Autoencoder in Intrusion Detection System. In Proceedings of the 2018 International Conference on Electrical Engineering and Computer Science (ICECOS), Pangkal, Indonesia, 2–4 October 2018; pp. 219–224.
20. Mhamdi, L.; McLernon, D.; El-moussa, F.; Zaidi, S.A.R.; Ghogho, M.; Tang, T. A Deep Learning Approach Combining Autoencoder with One-class SVM for DDoS Attack Detection in SDNs. In Proceedings of the 2020 IEEE Eighth International Conference on Communications and Networking (ComNet), Hammamet, Tunisia, 27–30 October 2020; pp. 1–6.
21. Chen, Z.; Yeo, C.K.; Lee, B.S.; Lau, C.T. Autoencoder-based network anomaly detection. In Proceedings of the 2018 Wireless Telecommunications Symposium (WTS), Phoenix, AZ, USA, 17–20 April 2018; pp. 1–5.
22. Elsayed, M.S.; Le-Khac, N.-A.; Dev, S.; Jurcut, A.D. Network Anomaly Detection Using LSTM Based Autoencoder. In Proceedings of the 16th ACM Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet '20), Alicante, Spain, 16–20 November 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 37–45.