IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 36, No. 5, October, 2022, pp. 737-743
Journal homepage: http://iieta.org/journals/ria

Stabbing of Intrusion with Learning Framework Using Auto Encoder Based Intellectual Enhanced Linear Support Vector Machine for Feature Dimensionality Reduction

Yadala Prabhu Kumar*, Burra Vijaya Babu

Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram-520002, Andhra Pradesh, India

Corresponding Author Email: prabhukumaryadala@gmail.com

https://doi.org/10.18280/ria.360511

Received: 8 August 2022
Accepted: 18 October 2022

Keywords:
network security, intrusion detection, learning framework, linear support vector machine, auto encoder, intellectual model

ABSTRACT

Using an Intelligent Intrusion Detection System (IIDS) instead of less effective firewalls and other intrusion detection systems can increase network security. The system's overall effectiveness is determined by the accuracy and speed of IIDS' categorization and training algorithms. According to research, Stabbing Intrusion Using Learning Framework (SILF) is an innovative and intelligent method of learning attack features and lowering dimensionality. To improve Enhanced Long Short-Term Memory (ELSTM) prediction accuracy while minimising testing and training time, an auto-encoder approach, which is an efficient learning methodology for feature generation in an unsupervised way is applied. Initial training samples are fed into the classifier to increase the predictability of incursion and classification accuracy. Thus, model efficacy may be achieved linearly while alternative classifier approaches such as conventional SVM, Random Forest (RF), and Naive Bayesian (NB) are investigated and compared. In this research, an efficient Intelligent Intrusion Detection System using Auto Encoder with Enhanced LSTM (IIDS-AE-ELSTM) is proposed for feature dimensionality reduction. Testing and training have shown that the proposed model is more efficient than existing systems in terms of performance measures such as accuracy, precision, recall, and F-measure. A new method to intrusion detection is presented, which increases detection of network intrusions with dimensionality reduction. The Python environment is used in the proposed model to create an efficient dimensionality reduction model for intrusion detection.

1. INTRODUCTION

Security breaches and assaults can be detected using intrusion detection systems (IDS). An ID system monitors and analyses network traffic in order to detect any irregularities. Signature-based or anomaly-based intrusion detection can be used [1]. Any deviation from the specified set of criteria is deemed as an intrusion according to the signature-based approach to finding intrusions [2]. This approach has high accuracy and low false alarm rates when recognising known assaults. This approach cannot identify a new type of assault due to the pre-determined rules. Analyzed intrusion detection systems can detect previously unknown or novel threats [3]. According to numerous research models, anomaly-based detection is more accurate, although it has a high percentage of false alarms. The lack of a supervised database for training machine learning models, as well as the difficulties in selecting an optimal feature from the network traffic dataset, make detecting intrusions more difficult. Because assault patterns change over time, it is difficult to discriminate between different types of attacks using the same features [4].

With the help of an IDS, businesses can keep tabs on their entire networks and more easily adhere to security standards. In addition, companies can utilise IDS logs as evidence of having complied with specific regulations. As an added bonus, intrusion detection systems can help authorities react more quickly to threats. Individuals, businesses, and governments all have a vested interest in ensuring that their networked systems are secure. Networked system attacks have expanded tremendously, and the methods employed by those behind them are constantly developing and adapting. One defence against these assaults is intrusion detection. An intrusion detection system's major use is in alerting IT staff to potential attacks or network intrusions. Any data travelling between computers on a network, as well as data coming into and going out of the network, can be spotted by an NIDS. The network IDS keeps an eye on everything going via the network and sends out alerts whenever something looks fishy or when recognised risks are found, allowing IT staff to investigate further and take preventative measures.

IDS are crucial for monitoring networks for malicious activity. An identification system listens for any deviations in the flow of traffic at the gates [5]. Intrusion detection can employ either a signature-based or anomaly-based approach. Intrusion detection systems have specified criteria that they utilise to identify and indicate anomalies in network traffic as potential intrusions [6]. With this method, you can reliably identify specific forms of assault with few false positives. As a result, this approach cannot be used to detect the new type of assaults as the stated principles do not correspond to the unknown patterns. Intrusion detection systems that have been thoroughly examined can identify brand-new or until unseen dangers. Theoretically, anomaly-based detection should be more trustworthy, however several published proofs show that

it has a large false alarm rate [7]. Important obstacles to intrusion detection include the lack of supervised data for training machine learning models and the challenge of choosing an optimal feature from the traffic flow dataset. It is challenging to differentiate between different sorts of attacks using the same attributes since attack patterns evolve over time.

The purpose of an IDS is to proactively monitor network traffic for malicious activity or attacks in progress. However, implementing IDS for massive, high-dimensional data streams is a formidable challenge. The efficiency of anomaly-based ID algorithms is heavily influenced by the fact that data streams contain features that are very different from those of statistics databases. These features comprise, but are not restricted to, the interactive nature of streaming data, the curse of dimensionality, small memory capacity, and high complexity, as well as the processing of massive data as it arrives (real-time). As a result, the primary issue in this field of study is to develop effective data-driven ID systems that can efficiently manage data streams while taking into account these unique aspects of traffic.

Given the massive size of modern datasets, it can be challenging to run numerous iterations of processing on the entire dataset at once to get reliable results. Processing or running an entire dataset is impacted significantly by training and computation time. It is feasible to solve this issue by reducing the problem's dimensionality. Once more, it is important to make sure you choose the right dimensionality reduction technique. It is challenging to identify a subset of attributes from an existing dataset that is useful or desirable, often leading to data loss or inappropriate criteria. Poor outcomes could be the result of using irrelevant features.

NIDS have recently been built using autoencoders, a deep-learning approach comprising of an encoder and a decoder [8]. The decoder uses the condensed latent vector as a starting point to recreate the original M-dimensional data from the compressed N-dimensional vector [9]. As long as the training error does not exceed a preset threshold, an input occurrence can be categorised as either normal or an attack [10]. An autoencoder-based NIDS can recognise patterns that deviate from established standards in order to detect new types of assaults. The process of dimension reduction of features is shown in Figure 1.

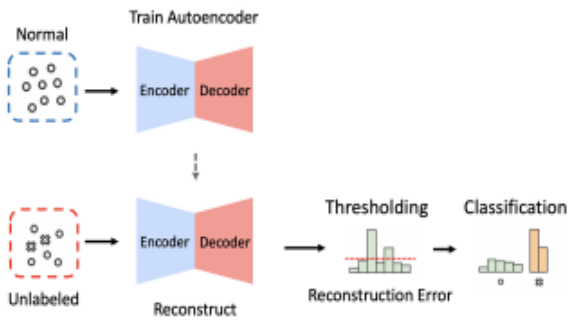

Figure 1. Process of dimension reduction

Artificial Neural Networks (ANN), SVMs, Naive Bayes classifiers, Random Forests (RF), and SOMs have all been utilised to develop an anomaly-based intrusion detection technique, as have RFs and RF random forests. They may be taught how to tell the difference between typical and abnormal traffic [11]. Anomaly-based techniques frequently use feature selection tasks to discriminate normal and abnormal traffic. During the feature selection phase, training data is condensed to a smaller dimension and redundant and irrelevant features are removed [12]. Meta-heuristic techniques and Principal Component Analysis (PCA) are just two of the tools available for narrowing down the ideal collection of qualities. The input data is compressed using a deep auto-encoder, and a dense neural network assesses atypical traffic [13].

The gathered traffic's attributes are compared to those of typical traffic, and any discrepancies are flagged by the anomaly-based intrusion detection system. Statistical learning, traditional machine learning, and deep learning are the three main categories of intrusion detection algorithms [14]. Statistical learning-based detection methods have been ruled out due to the requirement for data sharing. As AI has developed, machine learning systems have become more precise. Because of the data's multidimensionality and quantity, the data flow may be incredibly extensive and complicated [15]. When employing conventional machine learning algorithms, feature extraction must be performed by hand. Only very simple and superficial learning can profit from these methods. Artificial intelligence can be trained to automatically learn complex features from data using deep learning algorithms that construct a deep hierarchical network. By adopting the proposed method, new features can be generated rapidly [16].

In machine learning challenges, high-dimensional features contribute to lengthy categorisation processes [17]. Low-dimensional elements can be used to reduce the amount of time it takes to complete certain tasks. Because the classification of network traffic data contains unequal class distributions [18], most well-known classifiers, that assume usually balanced classification performance and equal miss-classification costs, have been considerably impeded [19]. Additional research is needed to solve the problem of uneven class distributions. An unequal distribution of classes in the classification of network traffic is something that past studies on intrusion detection systems have not addressed. There is also the possibility that the classifier's performance cannot be accurately assessed if the data is unbalanced [20].

The introduction section briefly discuss about the need of IDS and the significance of IDS using dimensionality reduction model [21]. The section 2 briefly provides the literature survey on various IDS models and the feature dimesionality reduction models, section 3 discuss the proposed model and the process of feature dimesionality reduction and section 4 presents the results section in which the proposed model is compared with the exisitng model and section 5 concludes the research.

2. LITERATURE SURVEY

The detection of network intrusions is presently being automated utilising expert networks and machine learning approaches. Because of the interconnection of multiple industrial control systems and the Internet environment, the Internet of Things (IoT) is vulnerable to cyber-attacks. Because of this, new machine learning methods with an emphasis on deep learning are being used to detect and categorise anomalies at the network and host levels. Dutta et al. [1] created an ensemble technique based on the stacking generalisation principle combining Deep Neural Networks and Long Short-Term Memory (LSTM), as well as meta-classifiers like regression models and DNN. The proposed

technology is able to detect anomalies more effectively using a two-step approach. During data pre processing, a Deep Sparse AutoEncoder (DSAE) is used to handle the feature engineering difficulty. In the second stage, categorisation is accomplished by the use of an ensemble learning approach.

IoT, Wireless Sensor Network (WSN), and modern computer networks all have to deal with network intrusion detection. Developing algorithms to detect network intrusions will be impossible without these datasets. Despite recent efforts, it is challenging to obtain datasets based on the actual real-world network traffic that may depict a wide range of different types of network attacks and invasions. LITNET-2020, the new labeled network benchmark dataset, was presented by Damasevicius et al. [3] in order to ease this necessity. The dataset serves as an example of typical and under-attack network traffic. Network flow features and 12 attack methods are included in the dataset, which the author explained and analyse in detail. The dataset's features are analysed using statistical and clustering techniques. The proposed feature set, based on research findings, is able to identify between distinct types of assault in the dataset.

Systems for detecting intrusions, malicious activity, or policy breaches in a computer system or network are known as intrusion detection systems. To keep up with today's ever-increasing volume and sensitivity of data, such systems are becoming increasingly important in today's networks. Intrusion detection is essential for IoT networks, which are becoming increasingly valuable as a target for intrusion attempts. Lopez-Martin et al. [4] proposed a new approach to network intrusion detection for IoT networks. The author proposed a conditional variational autoencoder for incorporating intrusion labels into the decoder layers. It's easier to use variational autoencoders for unsupervised classification, yet the proposed method provides superior classification results than other well-known classifiers. The method's capacity to rebuild features can be used to recover features from partial training datasets. Despite the vast number of possible values for categorical categories, the author demonstrated extremely good reconstruction accuracy.

Intrusion detection systems are critical to network security. There is a problem with traditional machine learning, however, because to the introduction of new assaults and uneven data sets. An improved conditional variational AutoEncoder (ICVAE)-DNN intrusion detection model was introduced by Yang et al. [9] for the first time. ICVAE is a tool for discovering and experimenting with network data representations that are sparse in structure. By producing new attack samples depending on defined intrusion categories, the ICVAE decoder's training data is rebalanced, resulting in improved diagnostic accuracy for imbalanced attacks. The weights of DNN's hidden layers can be fine-tuned using the ICVAE encoder that was previously learned, making it easier for DNN to perform global optimization via back propagation and fine tuning. The ICVAE-DNN is evaluated using the NSL-KDD and UNSW-NB15 datasets.

The Aho-Corasick algorithm (AC) and the Knuth-Morris-Pratt algorithm (KMP) is used to create, deploy, test, and evaluate a lightweight portable intrusion detection system (LPIDS). Consequently, this research adds to the following areas in three ways: To begin, an effective and lightweight IDS (LPIDS) is described by Nykvist et al. [13]. LPIDS was built with Aho-Corasick and KMP and tested on two different hardware platforms: The Raspberry Pi and the Wi-Fi Pineapple. The proposed LPIDS and its competitors are compared in terms of network properties such as throughput, energy consumption, and reaction time. Consultants are also invited to undertake security audits using the anticipated LPIDS. With fewer rules and a shorter learning curve, Aho-Corasick is generally speedier in the beginning of the process.

According to the analysis by Wang et al. [16], the optimal strategy for anomaly-base intrusion detection systems can be determined using a game-theoretic analysis method. A two-stage game model represents the attackers and defenders. In the early stages of a game, both defenders and attackers consider attack intensity and detection thresholds while selecting whether or not to engage in combat. An investigation of Nash equilibriums is conducted utilising six different scenarios, from which the optimal detection and attack actions can be inferred.

The use of IDS has grown in prominence as a crucial component of modern security infrastructure and a useful tool for risk management. We might think of an IDS as a type of pattern recognition system, and as such, feature extraction is a crucial part of the system's initial processing. The steps of feature creation and feature selection make up feature extraction. One of the most significant variables that determines an IDS's efficacy is the quality of the feature building and feature selection algorithms. The overall efficiency of the IDS can be greatly enhanced by achieving the goal of reducing the amount of relevant traffic features without sacrificing classification accuracy. Most intrusion detection effort, including feature building and feature selection, is still done manually, with the aid of domain knowledge. Filter, wrapper, and embedding approaches from machine learning are often used for autonomous feature creation and feature selection.

3. PROPOSED MODEL

The autoencoder method and its deep version have been very successful in traditional dimensionality reduction methods due to neural networks' powerful representability. They, however, do not capture the underlying functional manifold structure and instead use each instance to replicate itself rather than explicitly modelling the data relation [22]. This study reduced dimensionality through manifold learning by iteratively investigating data linkages and then leveraging those relationships to look for manifold structure [23]. A neural net's basic framework consists of an input layer, a hidden layer, and an outer layer. Deep neural networks are those with more than two hidden layers of connections [24]. Backpropagation, which takes advantage of the neural architecture's hidden layers, is used by neural networks to modify the weights and biases associated with a given neuron [25].

In a neural network, the autoencoder learns from its inputs and outputs how to be the same. The input can be reconstructed into output using two fundamental mathematical functions in the neural network. Both the encoding and decoding functions are given by x=f(g). Input g is translated to output p using the internal representation x. The network can learn the most useful aspects of the raw dataset by producing approximation replicas of input. Autoencoder neural networks have a bottleneck layer with fewer nodes than the other layers because the x is typically a smaller-dimensional subspace of the g.

As an unsupervised neural network, Autoencoder is capable

of rebuilding input vectors. Noise removal and intrusion detection techniques are just two examples where it exhibits impressive nonlinear generalisation skills. Here, the Autoencoder's hierarchy structure is shown in Figure 2. In an Autoencoder, the encoder and decoder are separate components. Encoders use a function named x=f0 to map input data to a feature space (g). It is useful to interpret the decoder as the function x=g0 for the generation and restructuring of the input vector (g).

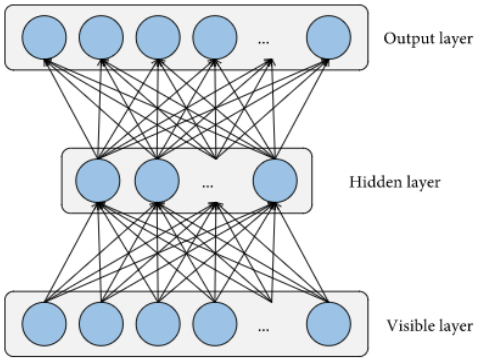

Figure 2. Structure of the autoencoder

In contrast to signature-based NIDS, autoencoder-based NIDS can identify previously undiscovered attack vectors by spotting deviations from regular traffic patterns. Autoencoders are useful for filtering out irrelevant information from datasets. Autoencoders let you narrow your focus to the most important aspects of your data by compressing it, encoding it, and afterwards recreating it as an output. To achieve certain effective techniques of training networks to acquire normal behaviour, autoencoders make use of a neural network's property in a unique way. As soon as an anomalous data point is introduced, the auto-encoder becomes ineffective. It picked up representations for patterns that weren't present in the data set.

A compressed representation of input from a high-dimensional space is encoded and is used by the decoder, as well as the input to the encoder is recreated using the compressed representation. Progress is made gradually in the tuning of encoder and decoder weights during the training phase. Reconstruction errors should be kept to a minimum. When a collection of training cases is given, a feature vector with dimension di can be calculated as X=(x1, x2,..., xm). An input vector of d dimensions is transformed into a hidden vector representation by the encoder layers of the Ae. An efficient Intelligent Intrusion Detection System using Auto Encode with Efficient LSVM is proposed for feature dimensionality reduction.

Algorithm IIDS-AE-ELSVM

{

Input: Intrusion Dataset {IDSET}

Output: Feature Subset Vector {FSV}

Step-1: The intrusion dataset is considered for analysis and the records are extracted from the dataset for deep analysis. The data set records loading and analysis is performed as:

$$RecV = \sum_{i=1}^{M} getValue\big(IDSET(i)\big) + getattr(i)\epsilon IDSET \quad (1)$$

Step-2: The artificial neural networks using ELSTM (linear long short-term memory) are common in intelligent machines and deep learning. ELSTM has feedback connections, unlike feed forward neural networks. Classifying and forecasting intrusions, both known and unknown, can be accomplished using ELSTM. The ELSTM components are initialized as

Forget gate→ "ft"

Candidate layer →"Ct"

Input gate→ "It"

Output gate→ "Ot"

Hidden state →"Ht"

Memory state →"Mt"

Forget gate:

$a_f = W_f.Z_t + b_f \qquad f_t = sigmoid(a_f)$

Input gate: $a_i = W_i.Z_t + b_i \qquad i_t = sigmoid(a_i)$

$: a_c = W_c.Z_t + b_c \qquad \tilde{c}_t = \tanh(a_c)$

Output gate: $a_o = W_o.Z_t + b_o \qquad o_t = \tanh(a_o)$

Cell state: $C_t = f_t \otimes C_{t-1} \oplus i_t \otimes \tilde{c}_t$

Hidden state: $h_t = o_t \otimes \tanh(c_t)$

Output equations: $V_t = W_v.h_t + b_t$

$\hat{y}_t = soft\max(V_t)$

$$L(i,j) = \frac{1}{V_t}\sum_{i=1}^{m}\|C_t - h_t\|^2 + W_c * W_0 \quad (2)$$

Step-3: The encoder and decoder functions are considered for performing auto encoding and decoding operations. The encoder and decoder sub-models make up an auto encoder. When the input is encoded, the encoder compresses it, and the encoder's decoder tries to re-create it from the encoded version. It is saved as an encoder model after training, and the decoder is thrown away after training. The encoding and decoding process is performed as:

$$E_k = f_\theta(g_c) = h\left(\sum_{i=1}^{m_1} W_i.b_i + h_i\right) + V_t - min(L(i,j)) + Mt \quad (3)$$

$$D_k = g_{\theta'}(f_c) = O\left(\sum_{i=1}^{m} y_i.c_i + b_t\right) - V_t + max(L(i,j)) - Mt \quad (4)$$

Step-4: Functions that define how the weighted sum of input is turned into an output from a layer of the network are known as activation functions. A neuron's activation function determines whether or not it will be triggered and passed on to the next layer, making it a critical component of a neural network. Just to be clear: This means that the network will make a judgement call about whether or not the neuron's input is significant to its prediction. The activation function is initiated as:

$$Af_i^{(W)} = f\big(Y_i^{(h)}\big) = f\left(\sum_{i=1}^{n} W_i^{(L-1)}.a_L^{(V-E)} + b_L^{(V-D)}\right) \quad (5)$$

Step-5: Feature extraction can be used to reduce the size of the data collection by removing extraneous information. That

means that the model can be built more rapidly and with less machine effort, allowing for faster progress through the training and generalisation phases. The feature extraction process is performed as:

$$FV_W = \frac{1}{2}\sum_{i=1}^{n} L(max(Af_i^{(W)}, Af_i^{(h)}) + min(C, V) \quad (6)$$

Step-6: Transformation of high-dimensional data into low-dimensional space in order to retain certain significant properties in the lower-dimensional representation, ideally near to its intrinsic dimension. The dimensionality reduction process is performed as:

$$Iset = \frac{FV^{(W)} - C[V^{(E)}]}{\sqrt{Var[W^{(L)}]}} \quad (7)$$

$$FSV[Iset^{(k)}] = \frac{1}{V}\sum_{i=1}^{n}\left(max(Iset_i^{(h)} - L[W^{(C)}]\right)^2 + max(Af_i, Af_{i+1}) \quad (8)$$

}

4. RESULTS

To discriminate between regular and abnormal traffic, anomaly-based approaches often employ feature selection tasks. Training data is reduced to a lower dimension and redundant features are eliminated during the feature selection process. Deep neural networks and signature-based methods are used to identify many different sorts of attacks in this area. The intrusion detection model incorporates data from the UNSW dataset as part of its training and evaluation. The experiment examined the proposed network's efficiency on the normal, R2L, U2R, and probe in binary and multi classification trials.

The proposed efficient Intelligent Intrusion Detection System using Auto Encode with Efficient LSTM (IIDS-AE-ELSTM) model is compared with the traditional SVM, Random Forest (RF), and Naive Bayesian (NB) models and the results represent that the proposed model performance is better.

By cleaning, altering, and modelling the data, analysts might uncover new insights that can help them make better decisions. The primary purpose of Data Analysis is to extract meaningful data and make a decision based on that knowledge. The intrusion data set is considered and the records are analysed for further processing. The data analysis accuracy levels are illustrated in the Figure 3.

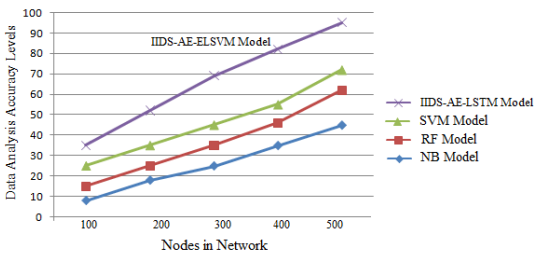

Figure 3. Data analysis accuracy levels

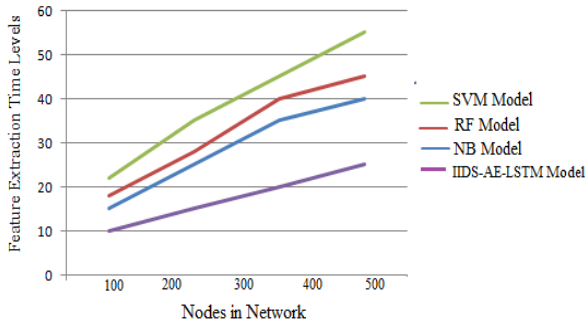

Figure 4. Feature extraction time levels

Feature extraction is the process of transforming raw data into numerical features that may be processed while maintaining all of the original information. When compared to machine learning alone, it is more effective. In order to identify all of the features that will be utilised to train the model, feature extraction is essential. The Figure 5 illustrates the feature extraction accuracy levels of the existing and proposed models.

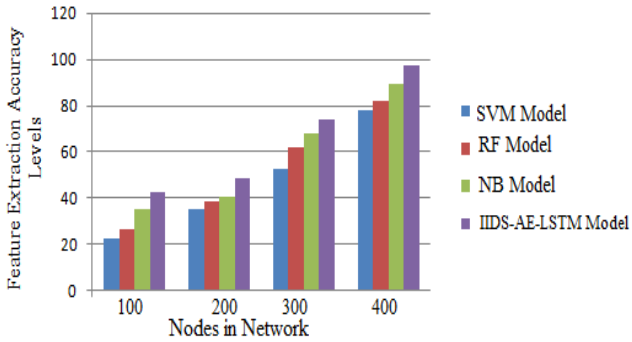

Figure 5. Feature extraction accuracy levels

Network traffic is monitored by IDS, which delivers alerts when it detects any suspicious activity. For instance, it is a piece of software that searches a network for suspicious behaviour or rules violations. An identity and access system is typically used to report any harmful activity or violation to an administrator, or to collect the data centrally. The stabbing of intrusion accuracy levels are shown in Figure 6.

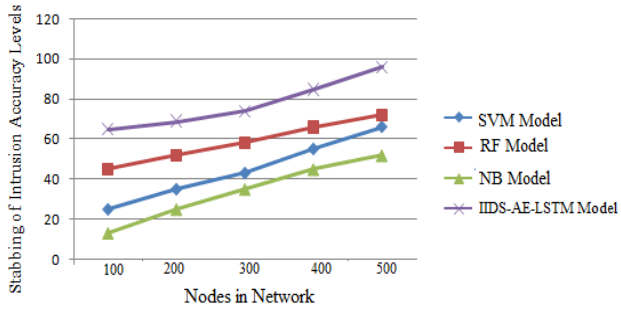

Figure 6. Stabbing of intrusion accuracy levels

Feature dimensionality reduction is able to reduce the amount of features in a resource-intensive computation without sacrificing vital information, which is known as dimensionality reduction. A reduction in features indicates a reduction in variables, which makes the computer's operation

more efficient and faster. The feature dimensionality reduction time levels of existing and proposed models are shown in Figure 7.

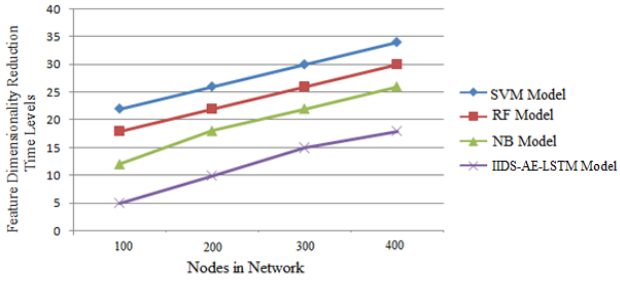

Figure 7. Feature dimensionality reduction time levels

Feature selection is the process of deciding which features to include and which to exclude. The process of reducing the size of an object's dimensions. To reduce the dimensionality of a dataset, the feature selection technique can be used to choose a set of attributes from a larger set. It is not possible to combine feature extraction and feature engineering. The feature dimensionality reduction accuracy levels of the proposed and existing models are shown in Figure 8.

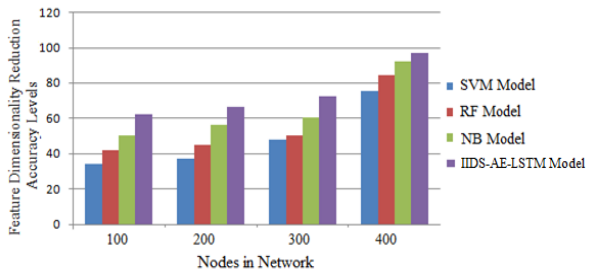

Figure 8. Feature dimensionality reduction accuracy levels

5. CONCLUSIONS

It is possible to translate high-dimensional data to a lower-dimensional format while still keeping the original geometry. Many different applications rely on it, including artificial intelligence (AI), content-based retrieval (CBR), bioinformatics, machine learning (ML), text analytics (TA), and intrusion detection (ID). Algorithms that analyse or identify patterns in the data require the repetition of comparison and distance computations inside the original data source. Machine learning operations like clustering and classification can be carried out at a lower dimensionality when similarity is maintained while dimensionality is reduced. AE and PCA models can be compared based on their reconstruction errors. The intrusion detection procedure is implemented with the help of two distinct levels of accuracy. In order to detect network anomalies, the proposed model employs a limited number of features, and if an anomaly is discovered, the same data sample is transmitted through a dense network. Using this strategy, a significant amount of data may be used to train binary and multiclass classifiers in real time. It has been proven that the deep AE model can recreate data points from a variety of distributions. An intrusion detection method that takes into consideration high-dimensional properties and data imbalance is proposed in this research. The framework's two primary components are feature dimensionality reduction and categorization. When doing feature dimensionality reduction, the 146-dimensional data features can be reduced to 18-dimensionally encoded data features using the proposed model. It might theoretically reduce the model's computational load. An efficient Intelligent Intrusion Detection System using Auto Encode with Efficient LSVM is proposed in this research for dimensionality reduction for intrusion detection. The proposed model achieves 98% accuracy in dimensionality reduction. Although the suggested methodology is capable of minimizing the features of a single big network, future improvements can be made by applying dynamic feature dimensionality. In the future, we will be able to use feature reduction, which can increase accuracy while decreasing the time complexity required. More training samples are needed to effectively identify the new forms of attacks.

REFERENCES

[1] Dutta, V., Choraś, M., Pawlicki, M., Kozik, R. (2020). A deep learning ensemble for network anomaly and cyber-attack detection. Sensors, 20(16): 4583.

[2] Parmisano, A., Garcia, S., Erquiaga, M.J. (2020). A labeled dataset with malicious and benign iot network traffic. Stratosphere Laboratory: Praha, Czech Republic. https://www.stratosphereips.org/datasets-iot23.

[3] Damasevicius, R., Venckauskas, A., Grigaliunas, S., Toldinas, J., Morkevicius, N., Aleliunas, T., Smuikys, P. (2020). LITNET-2020: An annotated real-world network flow dataset for network intrusion detection. Electronics, 9(5): 800.

[4] Lopez-Martin, M., Carro, B., Sanchez-Esguevillas, A., Lloret, J. (2017). Conditional variational autoencoder for prediction and feature recovery applied to intrusion detection in iot. Sensors, 17(9): 1967.

[5] Narayana, V.L., Sirisha, S., Divya, G., Pooja, N.L.S., Nouf, S.A. (2022). Mall customer segmentation using machine learning. In 2022 International Conference on Electronics and Renewable Systems (ICEARS), Tuticorin, India, pp. 1280-1288.https://doi.org/10.1109/ICEARS53579.2022.9752447

[6] Shahid, M.R., Blanc, G., Zhang, Z., Debar, H. (2019). Anomalous communications detection in IoT networks using sparse autoencoders. In 2019 IEEE 18th International Symposium on Network Computing and Applications (NCA), Cambridge, MA, USA, pp. 1-5. https://doi.org/10.1109/NCA.2019.8935007

[7] Pa, Y.M.P., Suzuki, S., Yoshioka, K., Matsumoto, T., Kasama, T., Rossow, C. (2015). IoTPOT: Analysing the rise of IoT compromises. In Proceedings of the 9th USENIX Workshop on Offensive Technologies (WOOT), Washington, DC, USA, 10-11 August.

[8] Ullah, I., Mahmoud, Q.H. (2020). A scheme for generating a dataset for anomalous activity detection in iot networks. In Canadian Conference on Artificial Intelligence, pp. 508-520. https://doi.org/10.1007/978-3-030-47358-7_52

[9] Yang, Y., Zheng, K., Wu, C., Yang, Y. (2019). Improving the classification effectiveness of intrusion detection by using improved conditional variational autoencoder and deep neural network. Sensors, 19(11): 2528.

[10] Sheetal, A.P., Lalitha, G., Gopi, A.P., Narayana, V.L. (2021). Secured data transmission with integrated fault reduction scheduling in cloud computing. Ingenierie des Systemes d'Information, 26(2): 225-230. https://doi.org/10.18280/isi.260209

[11] Patibandla, R.S.M., Narayana, V.L. (2021). Computational intelligence approach for prediction of COVID-19 using particle swarm optimization. In Computational Intelligence Methods in COVID-19: Surveillance, Prevention, Prediction and Diagnosis, pp. 175-189. https://doi.org/10.1007/978-981-15-8534-0_9

[12] Kingma, D.P., Ba, J. (2015). Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7-9 May.

[13] Nykvist, C., Larsson, M., Sodhro, A.H., Gurtov, A. (2020). A lightweight portable intrusion detection communication system for auditing applications. International Journal of Communication Systems, 33(7): e4327. https://doi.org/10.1002/dac.4327

[14] Radoglou-Grammatikis, P.I., Sarigiannidis, P.G. (2019). Securing the smart grid: A comprehensive compilation of intrusion detection and prevention systems. IEEE Access, 7: 46595-46620. https://doi.org/10.1109/ACCESS.2019.2909807

[15] Thakkar, A., Lohiya, R. (2020). A review of the advancement in intrusion detection datasets. Procedia Computer Science, 167: 636-645. https://doi.org/10.1016/j.procs.2020.03.330

[16] Wang, Z., Xu, S., Xu, G., Yin, Y., Zhang, M., Sun, D. (2020). Game theoretical method for anomaly-based intrusion detection. Security and Communication Networks, 2020: 8824163. https://doi.org/10.1155/2020/8824163

[17] Altwaijry, N., ALQahtani, A., AlTuraiki, I. (2019). A deep learning approach for anomaly-based network intrusion detection. In International Conference on Big Data and Security, pp. 603-615. https://doi.org/10.1007/978-981-15-7530-3_46

[18] Kwon, D., Kim, H., Kim, J., Suh, S.C., Kim, I., Kim, K.J. (2019). A survey of deep learning-based network anomaly detection. Cluster Computing, 22(1): 949-961. https://doi.org/10.1007/s10586-017-1117-8

[19] Dong, Y., Wang, R., He, J. (2019). Real-time network intrusion detection system based on deep learning. In 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS), pp. 1-4. https://doi.org/10.1109/ICSESS47205.2019.9040718

[20] Althubiti, S.A., Jones, E.M., Roy, K. (2018). LSTM for anomaly-based network intrusion detection. In 2018 28th International telecommunication networks and applications conference (ITNAC), pp. 1-3. https://doi.org/10.1109/ATNAC.2018.8615300

[21] Radoglou-Grammatikis, P.I., Sarigiannidis, P.G. (2018). An anomaly-based intrusion detection system for the smart grid based on cart decision tree. In 2018 Global Information Infrastructure and Networking Symposium (GIIS), pp. 1-5. https://doi.org/10.1109/GIIS.2018.8635743

[22] Senthilnayaki, B., Venkatalakshmi, K., Kannan, A. (2019). Intrusion detection system using fuzzy rough set feature selection and modified KNN classifier. The International Arab Journal of Information Technology, 16(4): 746-753.

[23] Salo, F., Nassif, A.B., Essex, A. (2019). Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection. Computer Networks, 148: 164-175. https://doi.org/10.1016/j.comnet.2018.11.010

[24] Sun, J., Wang, X., Xiong, N., Shao, J. (2018). Learning sparse representation with variational auto-encoder for anomaly detection. IEEE Access, 6: 33353-33361. https://doi.org/10.1109/ACCESS.2018.2848210

[25] Ali, S., Li, Y. (2019). Learning multilevel auto-encoders for DDoS attack detection in smart grid network. IEEE Access, 7: 108647-108659. https://doi.org/10.1109/ACCESS.2019.2933304