Enhancing Threat Detection in Financial Cyber Security Through Auto Encoder-MLP Hybrid Models

Layth Almahadeen[1], Ghayth ALMahadin[2], Kathari Santosh[3],
Mohd Aarif[4], Dr. Pinak Deb[5], Maganti Syamala[6], Dr B Kiran Bala[7]

Lecturer, Department of Financial and Administrative Sciences, Al- Balqa' Applied University, Jordan[1]
Assistant Professor, Department of Networks and Cybersecurity-Faculty of Information Technology,
Al Ahliyya Amman University, Jordan[2]
Assistant Professor, Department of MBA, CMR Institute of Technology, Bengaluru, India[3]
Department of Commerce, Aligarh Muslim University, Aligarh, Uttar Pradesh, India[4]
Assistant Professor, Department of MBA-Sanjivani College of Engineering, Savitribai Phule Pune University, Pune, India[5]
Assistant Professor, Dept. of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation,
Vaddeswaram, Guntur Dist., Andhra Pradesh, India[6]
Head of the Department, Department of Artificial Intelligence and Data Science, K.Ramakrishnan College of Engineering,
Trichy, Tamil Nadu, India[7]

***Abstract*—Cyber-attacks have the potential to cause power outages, malfunctions with military equipment, and breaches of sensitive data. Owing to the substantial financial value of the information it contains, the banking sector is especially vulnerable. The number of digital footprints that banks have increases, increasing the attack surface available to hackers. This paper presents a unique approach to improve financial cyber security threat detection by integrating Auto Encoder-Multilayer Perceptron (AE-MLP) hybrid models. These models use MLP neural networks' discriminative capabilities for detection tasks, while also utilizing auto encoders' strengths in collecting complex patterns and abnormalities in financial data. The NSL-KDD dataset, which is varied and includes transaction records, user activity patterns, and network traffic, was thoroughly analysed. The results show that the AE-MLP hybrid models perform well in spotting possible risks including fraud, data breaches, and unauthorized access attempts. Auto encoders improve the accuracy of threat detection methods by efficiently compressing and rebuilding complicated data representations. This makes it easier to extract latent characteristics that are essential for differentiating between normal and abnormal activity. The approach is implemented with Python software. The recommended Hybrid AE+MLP approach shows better accuracy with 99%, which is 13.16% more sophisticated, when compared to traditional approach. The suggested approach improves financial cyber security systems' capacity for prediction while also providing scalability and efficiency while handling massive amounts of data in real-time settings.**

Keywords*—Financial cyber security; auto encoder; multilayer perceptron; threat detection; hybrid models

I. INTRODUCTION

The necessity of cyber safety and defense against various forms of cyber-attacks has increased dramatically in the last several years. The phrase "cyber security" describes a group of policies, mind-sets, and behaviours that help safeguard electronic data. Cyber-attacks including computer viruses [1], DoS attacks [2], and unlawful access have caused irreversible harm and financial harms in massive networks. For instance, a single ransom ware infection cost $8 billion in significant damages to several industries and enterprises, including banking, energy, healthcare, and higher education. Global investment in cyber security is expected to reach $1 trillion by 2021; in 2013, spending increased by more than 40% to $66 billion. Lately, cyber security researchers have started looking at AI techniques to improve cyber security. Similarly, fraudsters are using AI to launch increasingly sophisticated attacks while avoiding detection. However, we focus on how AI-powered cyber security solutions may lessen or completely prevent data breaches and more effectively fight attackers in our work [3]. AI has come a long way since it was first developed in the 1950s, yielding many interesting systems and research discoveries. ML and DL were the products of further developments. AI is being employed these days in many different domains, including industry, law, and exploration of space, medical care, and agriculture. New paradigms such as cloud-based computing and big data, along with continuous improvements in computer hardware and software performance and decreasing costs, have made it easier to develop and deploy a wide variety of AI systems with varying skills [4].

These days, a lot of these AI systems are capable of carrying out a wide range of difficult tasks, such as face and speech recognition, planning, problem solving, and learning [5]. Another important development in AI since the 1980s has been the emergence of technologies for ML, which allow machines to learn and adapt to various environments by utilizing their past experiences, patterns, and knowledge. The field of ML, came into being ten years ago. With the help of this sector, robots may uncover latent correlations in the information they are given, improving planning and forecast accuracy. Recently, there has been an increase in interest in applying AI and ML techniques to counter cyber-attacks [6]. The usage of these technologies is mostly driven by the vast amounts of information that are being created, since they need a significant time and resource commitment to analyze and detect any trends, irregularities, or breaches in traffic data.

The terms "cyber banking" and "cyber security" describe protocols, practices, and infrastructures that protect data, networks, and computer programs from online threats [7]. The threat posed by cyber security is one kind of financial terrorism which has become more prevalent. The most challenging aspect of modern cyber banking has shown to be the protection of customers' personal information. Cyber security is a strategy for thwarting cyber-attacks in cyberspace. Theft of confidential data, including account and ID numbers, and private information are examples of non-financial losses [8]. Cyber security aims to shield the impacted company and its customers from the monetary and non-monetary damages that result from a breach in any kind of data security system. Cybercrime has a detrimental financial effect on South African communities and impacts the whole planet. Safeguarding sensitive data is one of the most significant concerns of cyber security and confidentiality in the realm of cyber banking.

Technology breakthroughs have brought about changes in the banking sector, with internet banking developing as a more sensible method of conducting business. South African banks frequently employ third-party services like PayPal for both domestic and international transactions. Since the banks have no influence over the administration of these systems, their dependence on outside vendors to guarantee the caliber of their online offerings for clients poses a significant security risk. System connection promotes reliance, but it also raises the risk of cyber-attacks and breaches. Managing these risks means preventing and lessening assaults before they occur is termed as risk management. Banking was disproportionately affected by a 1318% rise in ransom ware attacks in the initial half of 2021. The four percent increase in business email compromise, or BEC, assaults might be attributed to new COVID-19 options for threat actors. Large-scale cyber-attacks are becoming more and more likely to target banks. Because banks are linked, a cyber-attack on one might put the solvency of a financial institution at risk. Cyber-attacks on US banks that are supported by states are especially dangerous. As more individuals utilize the web and mobile banking, cybercrime has been rising over time. Cybercrime occurrences include a variety of fraud types, such as identity theft, ATM robberies, and credit card frauds. The banking sector is especially vulnerable because of the significant monetary worth of the information it holds. Hackers may make money in a variety of ways using the financial data and banking credentials they have taken. The attack surface available for exploitation has increased in tandem with the size of banks' digital footprints. Cyber-attacks have the potential to result in confidential information breaches, power disruptions, and malfunctioning military equipment. They may result in the theft of private information that can be quite valuable, including medical records. They have the ability to paralyze systems or interfere with computer and phone networks, making data inaccessible. Banking is especially vulnerable as the data it stores has significant value.

By combining the benefits of supervised and unsupervised learning methods, the suggested strategy improves the identification of threats in financial cyber security. Conventional approaches frequently find it difficult to keep up with the constantly shifting characteristics of cyber threats, particularly in the ever-changing financial industry. The article presents a novel framework that combines the discriminative strength of MLP [9] networks with the feature learning abilities of auto encoders in order to handle this difficulty. Our goal is to increase detection accuracy by utilizing labelled information and capturing intricate patterns in the data through the integration of these two methods into a hybrid model. This method not only makes it easier to spot existing hazards, but it also gives you the flexibility to spot new irregularities and questionable activity in financial systems. Using an auto encoder-MLP hybrid model, the research can leverage labelled data to refine the model for particular threat detection tasks while also efficiently extracting high-level descriptions of the basic data structure via unsupervised learning. With the help of this hybrid architecture, we can use unsupervised feature learning to take use of the inherent qualities of the data, strengthening the model's resistance to new and unknown threats. Additionally, the framework can improve overall threat detection performance by learning to discriminate between benign and harmful actions with better accuracy. Our suggested strategy provides a more flexible and effective protection against new threats, offering a viable answer to the cyber security concerns in the financial arena through the creative integration of supervised and unsupervised learning approaches.

The key contribution of the proposed Auto encoder-MLP hybrid models’ study is as follows

- The study suggests a novel approach to improve financial cyber security threat detection. By combining a Multilayer Perceptron (MLP) with an Auto Encoder (AE) this hybrid approach efficiently detects vulnerabilities within financial systems by utilizing the advantages of both constituent parts.
- By using min-max normalization to lessen the influence of feature size fluctuations, the study highlights the significance of data pre-processing in financial cyber security threat identification. This guarantees steady scalability and boosts model training effectiveness, which in turn raises threat identification accuracy.
- The architecture that has been suggested clearly outlines the functions of every element, ranging from feature extraction to threat detection in financial cyber security, hence promoting an open and effective framework for the creation and use of models.
- The proposed AE-MLP hybrid model is extensively tested using the NSL-KDD dataset, showing good results on a number of metrics including f1-score, recall, accuracy, and precision. The study offers in-depth understandings of the effectiveness and dependability of the suggested strategy, confirming its viability for practical implementation in financial cyber security environments.

The article's remaining sections are arranged as follows: A summary of relevant studies is given in Section II. The issue statement for the existing system is found in Section III. In

Section IV of the study, the proposed Auto encoder-MLP hybrid model and technique for increased threat detection are described. The study's results and the ensuing discussion are presented in Section V. Section VI discusses the suggested model's conclusion and possible applications.

II. Related Works

The paper by I. H. Sarker, Y. B. Abushark, F. Alsolami, and A. I. Khan [10] introduced the "IntruDTree," a ML-based security framework that builds a generalized detection of intrusion model based on a tree structure by first considering the importance ranking of security features. This approach lowers computation complexity for yet-to-be-tested test scenarios while retaining prediction accuracy by shrinking the feature dimensions. Lastly, by doing tests using cyber security datasets and evaluating metrics to assess, the efficacy of our IntruDTree model was investigated. To assess the efficacy of the resultant security model, the study also compare the outcomes of the IntruDTree model with a number of conventional, well-known ML techniques, including the naive Bayes classification algorithm, logistical regression, SVMs, and the k-nearest-neighbour model. The drawback of the model is static dataset is utilized to trained the model and relies on predetermined feature importance rankings, it may struggle to adapt quickly to new types of intrusions or novel attack patterns. Without regular retraining and upgrades, static models like the IntruDTree could discover it difficult to keep up with new threats as they emerge and get more complex over time.

IDS that utilizes a stacked AE and a DNN is proposed in the G. Muhammad, M. S. Hossain, and S. Garg [11] paper. In order to reduce the feature width, the stacked AE analyses the distinctive characteristics of the input networks recording in an unsupervised way. Subsequently, supervised training is applied to the DNN in order to obtain deep learning characteristics for the classifier. The DNN contains two or three layers in the suggested system, with each layer having a fully linked layer, a batch normalization layer, and a dropout. The AE has two latent layers. Three sets of publicly available data were used to assess the system. The trials' findings demonstrated the 94.2% accuracy of the recommended IDS for multiclass categorization. The disadvantage is that it has been demonstrated that adversarial examples carefully constructed inputs intended to distort the model's predictions can weaken DL models, particularly DNNs. Since the AE and DNN in the proposed IDS rely heavily on learned representations of network traffic data, they may be susceptible to adversarial manipulation of input features, leading to misclassifications or incorrect intrusion detection decisions.

The goal of the M. Alsaedi, F. A. Ghaleb, F. Saeed, J. Ahmad, and M. Alasl [12] study is to increase the detection accuracy of harmful URLs by creating a model for two-stage ensemble learning that is based on cyber threat intelligence. To increase detection accuracy, online searches are used to extract the attributes based on cyber threat data. Global user reports and cyber security analysts can offer vital information about rogue websites. Consequently, to enhance detection efficiency, characteristics derived from searches on Google and Whois websites are utilized to create cyber threat intelligence (CTI). The study also suggested a two-stage ensemble learning approach that combines multilayer perceptrons (MLP) for ultimate decision-making with the RF algorithm for pre classification. The three separately trained random forest classifiers' majority voting system has been superseded by the trained MLP classifier for decision-making. For sufficient classification, the probabilistic outputs of the random forest's weak classifiers was combined and fed into the MLP classifier. The retrieved CTI-based characteristics based on the two-stage classification perform better than the detection models used in other research, according to the results. Compared to the conventional URL-based model, the suggested CTI-based detection model produced a 7.8% accuracy gain and a 6.7% decrease in false-positive rates. The drawback of the model is the reliance on online searches for feature extraction may introduce biases in the dataset used for training the ensemble learning model. The model's performance could be impacted if the extracted attributes do not adequately represent the diversity of malicious URLs or if there are inherent biases in the online sources used for data collection.

An efficient methodology for detecting intrusions using SVM and naive Bayes feature embedding was presented by J. Gu and S. Lu [13] in their study. The naïve Bayes transform feature technique is used to build new, high-quality information from the original features; an SVM classifier is subsequently trained with the altered data to generate an ID model. Research was out on several datasets within the intrusion identification domain substantiate the efficaciousness and resilience of the recommended detection technique. The UNSW-NB15 dataset shows 93.75% accuracy, the CICIDS2017 dataset shows 98.92% accuracy, the NSL-KDD dataset shows 99.35% accuracy, and the Kyoto 2006+ dataset shows 98.58% accuracy. Furthermore, our method offers notable advantages over existing methods in terms of efficiency, false alarm rate, and identification rate. Keeping the scalability and effectiveness of the IDS is a difficulty when expanding the study to scenarios with varying forms of attacks. The feature space may grow dramatically as attack types become more sophisticated and diverse, increasing processing costs and perhaps reducing the model's capacity for real-time detection.

For the IIoT wireless sensing scenario, a DL-based network intrusion identification and categorization model (NIDS-CNNLSTM) is created in the J. Du, K. Yang, Y. Hu, and L. Jiang, [14] goal is to effectively distinguish and recognize network traffic while ensuring the equipment and operation of the IIoT are secure. Using LSTM in data from time series together with the powerful capacity for learning of neural networks, NIDS-CNNLSTM trains and classifies the features selected by the CNN and verifies its application through binary categorisation and multi-classification scenarios. The precision rate while categorizing different forms of traffic is high, and the three datasets show outstanding convergence and level in terms of validation accuracy, training loss, and precision rate. Previous study models were not able to equal NIDS-CNNLSTM's overall effectiveness. The experimental findings show good classification accuracy, a small false alarm rate, and a high detection rate. It is more appropriate for large-scale, multi-scenario network data in the IIoT. The primary disadvantage is that deep learning models, such as CNN-LSTM, can be computationally demanding, particularly when working with high-dimensional, large-scale data sets like network traffic data.

There are numerous types of security frameworks and IDS that have limitations. First, because some models rely on predefined feature rankings and static datasets, they are noticeably less flexible than others. This constraint limits their capacity to quickly adapt to novel forms of intrusions or developing assault patterns, which may risk their efficacy in quickly changing environments including cyber threats. Second, adversarial examples, complex inputs purposefully created to distort the model's predictions can affect deep learning-based IDS. This issue is quite concerning since it might result in incorrect intrusion detection judgments or misclassifications, which could compromise the system's overall dependability. Furthermore, algorithms that extract features from web searches might introduce biases into the training dataset. The model's performance may be impacted by this reliance on outside sources for feature extraction, which might lead to inadequate representation of the variety of harmful activities. Finally, there are still issues with scalability, especially when dealing with high-dimensional datasets and different kinds of attacks. Certain intrusion detection systems may encounter difficulties in maintaining scalability and efficiency when attack complexity escalates, which might result in increased computing expenses and reduced real-time detection capabilities. All of these drawbacks highlight the continuous requirement for enhanced threat detection system that can strike a balance between flexibility, dependability, and computing efficiency in the ever-changing world of cyber security threats.

III. Problem Statement

The financial cyber security environment faces a variety of challenges as a result of the deficiencies of current intrusion detection systems and security standards. These challenges include scalability issues with high-dimensional datasets and diverse attack vectors, biases introduced by algorithms extracting features from web searches, inherent inflexibility resulting from reliance on predefined features and static datasets, and vulnerability to adversarial examples in deep learning-based intrusion detection systems [15]. It is imperative to design a more sophisticated threat detection system that strikes a balance between dependability, adaptability, and computing efficiency in order to counteract these shortcomings and keep up with the constantly evolving landscape of cyber threats affecting the financial industry. In order to overcome these obstacles, this study suggests a novel strategy that makes use of Auto Encoder-Multilayer Perceptron (AE-MLP) hybrid algorithms. The goal is to provide a strong framework for threat detection that can efficiently detect and mitigate cyber risks in financial systems while preserving scalability, resilience to adversarial assaults, and flexibility.

IV. Proposed Auto Encoder-MLP Hybrid Model for Enhancing Threat Detection in Financial Cyber Security

In order to enhance threat detection in financial cyber security, this methodology suggests combining the use of an Auto Encoder (AE) with a Multilayer Perceptron (MLP). Due to their sensitive data, financial companies are particularly vulnerable to cyber-attacks. The approach begins by normalizing the data in order to guarantee consistent scalability in order to address this. The two primary components of the hybrid model are the AE, which compresses and extracts significant patterns from the data, and the MLP, which categorizes threats using this compressed representation. The number of layers, units per layer, activation functions, and other parameters must be specified in the MLP construction algorithm. All things considered, this technique provides a transparent framework for creating and implementing a hybrid AE-MLP model for improved threat identification in financial cyber security. Fig. 1 shows the block diagram of this AE-MLP methodology is given below.

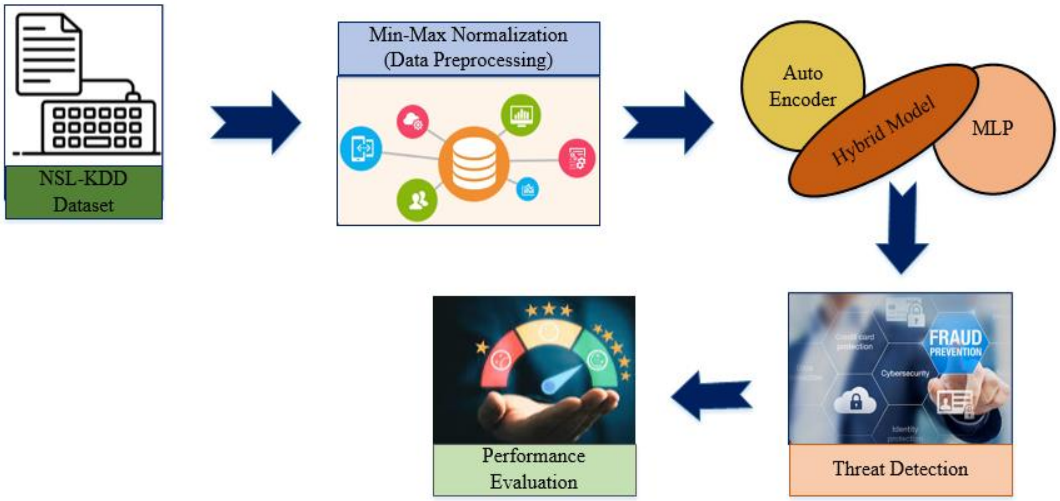

Fig. 1. Hybrid AE-MLP model block diagram.

A. Dataset Collection

NSL-KDD dataset was collected from the secondary source [16]. It consists of specific entries from the data collection KDD 99. The train and test sets contain identical records, and because of the decreased dataset size, random selection is not required. There exists a negative correlation between the proportion of entries in the KDD99 dataset and the chosen entries in every single category of the NSL-KDD dataset. Various ML algorithms may achieve a wider range of accuracy, resulting in more accurate assessments of various models. There are 125,970 occurrences in the training dataset and 225,440 samples in the test dataset. There are four categories into which the assaults are divided: DoS, R2L, U2R, Probe, and a Standard class.

B. Min-Max Normalization for Data Pre-processing

Normalization minimizes the impact of feature scale variations, which reduces the training time of a model. The min-max normalization is used once the outliers have been relocated. Mathematical data can be transformed into a range, often between 0 and 1, using a technique called min-max normalization, sometimes referred to as features scaling. All of the dataset's features, or columns, go through this procedure[17]. Scaling numerical data within a particular range, usually between 0 and 1, is known as min-max normalization. It is a fundamental data preparation method utilized in threat detection in the sets of data given using a hybrid Auto Encoder- MLP model. In the absence of an explicit calculation, this technique guarantees that the lowest and greatest values in the dataset are converted to 0 and 1, respectively, and that any further data values are adjusted linear with respect to this range [18]. The normalization process modifies each data point individually by computing the characteristic or column's lowest and highest values, removing the smallest value, and divided by the range of values. By ensuring uniform feature scaling, min-max normalization contributes to improved convergence and model stability, thereby enhancing the performance of auto encoder-MLP hybrid models. This enhancement is particularly valuable in threat detection scenarios, where model accuracy and robustness are paramount. The normalization of min-max is expressed using Eq. (1) and Eq. (2).

$$N_{std} = \frac{N - N_{min}}{N_{max} - N_{min}} \tag{1}$$

$$N_{scaled} = N_{std} \times (max - min) + min \tag{2}$$

By doing this, you can be sure that the values that fall between will be scaled linearly to match the transformation of the lowest value to 0 and the highest value to 1.This normalization method is particularly useful when features have different scales since it ensures uniformity among the features and supports the performance of the ML model during training.

C. Synergistic Auto Encoder-MLP Architecture for Advanced Threat Detection in Financial Cyber Security

The sensitive data that financial institutions hold makes them easy targets for cyber-attacks. To protect assets and preserve confidence in the financial system, financial cyber security threat detection must be improved. Deep learning methods have showed promise in identifying and reducing cyber dangers in recent years. In order to increase threat detection in financial cyber security, the study provide a hybrid model in this proposal that combines an auto encoder with a MLP.

An auto encoder is a type of multilayer neural network where the desired output is comparable to the input with less modifications, i.e., the result is similar as the inputs with some reconstruction error [19]. By encoding the input, the auto encoder uses unsupervised learning to decode or rebuild the output. Auto encoders are commonly used in recommender systems to decrease the dimensionality of characteristics, retrieve pertinent characteristics, compress and remove noise from the pictures, forecast sequences, and identify abnormalities.

For the purpose of conciseness, we describe the overall architecture of an auto encoder without getting into specifics.

A general auto encoder consists of four key components: the encoder, reconstruction loss, bottleneck, and decoder. The encoder shrinks the data into an encoded form and helps to reduce characteristics from the input. The layer with the fewest features and compressed incoming data is known as the bottleneck layer. By assisting the model in reconstructing the result from the encoded representation, the decoder ensures that the output and input are identical. Reconstruction Loss is the last term used to assess the decoder's performance and gauge how close the output is to the original input.

Moreover, back propagation is used to carry out training and reduce reconstruction loss even more. This minimum loss illustrates the objective that AE strives to achieve. The input y will be compressed by the encoder is expressed in Eq. (3)

$$x = E(y) \tag{3}$$

The input will be attempted to be recreated by Decoder D as y' = D (E (y)).

$$loss(E, D) = \frac{1}{n} \sum_{j=1}^{n} y^i - D(E(y^i)))^2 \tag{4}$$

The variation between the decoded and encoded vectors in this case is the reconstruction loss. One way to calculate the reconstruction loss is to use the Mean Square Error (MSE). It is provided in eqn. (4) given above. Fig. 2 shows the architectural diagram of hybrid AE-MLP is given below.

One kind of ANN that forms the basis of DL models is the MLP. Since MLPs belong to the class of feed forward neural networks, data moves from the input layer to the output layer only in one direction. For many different ML tasks, such as feature learning, regression, and classification, they are extensively utilized. Let's take a closer look at the parts, construction, and operation of an MLP [20]. An MLP is made up of several layers of linked neurons, and its structure is typified by three primary kinds of layers.

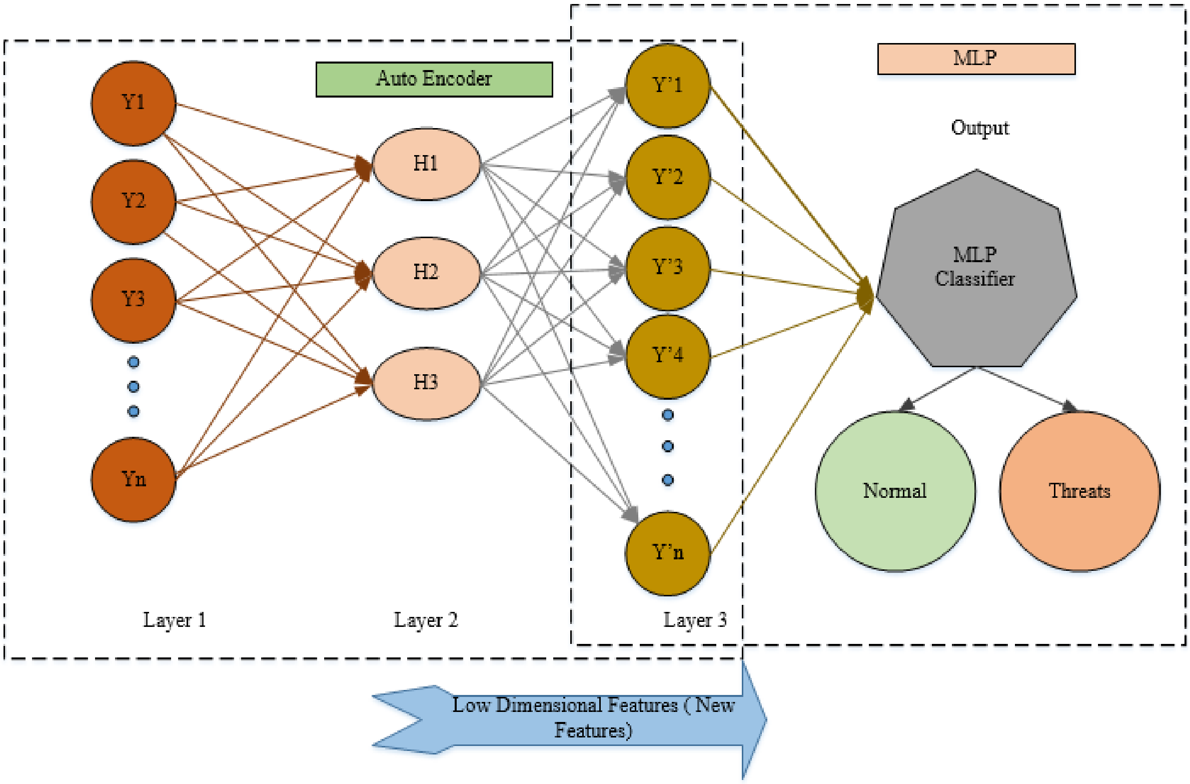

Fig. 2. Hybrid auto encoder-MLP architecture.

The first layer of the network's architecture is the information input layer, which gets raw input data. Since each neuron in the input layer is mapped to a feature in the dataset, the input layer serves as a fundamental demonstration of the data's measurements. The concealed layer comes next. Among the input layer and the output layer of an MLP, there may be several layers that are hidden. Because these layers are not immediately linked to the external world, they are referred to as "hidden" layers. Hidden layer neurons apply a function of activation to the weighed total of the inputs from the preceding layer, process the data, and then forward the output to the subsequent layer. One may modify the hyper parameters, such as the number of neurons and layers that are hidden in each layer, to maximize the models.

The output layer, which is the last layer, generates the prediction or model's output. The problem that the MLP is intended to address dictates this layer's construction. Multiclass classification may employ many neurons, each representing a class and utilizing a softmax activation function, in contrast to binary categorization, which would utilize a single neuron with a sigmoid activation function.

The neurons of an MLP perform the following functions.

- Weighted Sum: The input for every neuron is equal to the weighted total of the outputs from the layer that came before it. A common term for this weighted total is the neuron's "activation." The sum $y_q^{(l)}$of neuron q in layer l may be written mathematically as in Eq. (5)

$$y_q^{(l)} = \sum_p w_{pq}^{(l)}\, a_p^{(l-1)} + b_q^{(l)} \qquad (5)$$

Where,

$y_q^{(l)}$ Corresponds to the layer l activation of neuron q.

$w_{pq}^{(l)}$ The connection weight between neuron p in layer l−1 and neuron q in layer l is measured.

$a_p^{(l-1)}$ Is layer l−1 neuron p's output.

$b_q^{(l)}$ Is layer l's neuron q's bias.

- Activation Function: Common activation functions include the sigmoid function, tanh, and ReLU. One layer's output serves as the subsequent layer's input. The Eq. (6) denotes it.

$$a_p^{(l)} = f\,(y_q^{(l)}) \qquad (6)$$

where, $a_p^{(l)}$ is the layer l output of neuron q, and the activation function is represented by f.

- Feed Forward Propagation: Applying the activation functions to the weighted aggregate, the activations are computed for each of the neurons in the feed forward process. By using the final result of a specific layer as the input of the next layer, information is transferred from the layer that provided the input to the output layer. $a_p^{(l)}$ Is expressed in Eq. (7) is given below

$$a_p^{(l)} = (\sum_p w_{pq}^{(l)}\, a_p^{(l-1)} + b_q^{(l)}) \qquad (7)$$

To improve threat detection in financial cyber security, a hybrid model that combines the strong abilities of AE and

MLPs is suggested. As a skilled feature extractor in this design, the auto encoder is able to identify important patterns and representations that are hidden in the input data. The auto encoder uses its encoder network to generate a condensed space of latent information representation that captures the key elements of the input data. This representation of latent space is then sent into the hybrid model's MLP component. In its capacity as a classifier, the MLP uses the characteristics that it has extracted from the auto encoder to identify and group different threat categories that are present in the financial security space.

The hybrid model uses labelled datasets for supervised learning during the training phase. The AE and MLP elements must be simultaneously optimized in this combined training method. The MLP is simultaneously trained to reduce classification errors by utilizing the informative features obtained from the AE's latent space representation, while the auto encoder attempts to properly recreate the input data. Selecting a suitable loss function, such cross-entropy, makes it easier to quantify the difference between the expected and real labels, which helps to increase the model's resilience and classification accuracy.

Algorithm for MLP Architecture

```
Require: $D$train, $D$test (Training and Testing)
Require: MLP architecture hyper parameters

Data Pre-processing: Min-max normalization

Build the MLP model:
Layer Numbers : 3
Units/ layer:
Input layer: data containing number of features
Hidden layer 1: ReLU activation 128 units
Hidden layer 2: ReLU activation 64 units
Output layer: The quantity of output classes that possess an appropriate activation function

Assemble the MLP model by specifying the optimizer, loss, and assessment metrics.
For a certain number of epochs, train the MLP model on $D$train.
Determine performance indicators and assess the model on the $D$test.
Fine tune hyper parameters as needed
Optionally deploy the model
Keep monitor model and updated.
```

V. Results and Discussion

Using the NSL-KDD dataset, AE-MLP Threat detection approach is evaluated. The proposed approach produces excellent and extremely promising outcomes. Using a device operating Python as a programming language with the Windows 10 operating system. The performance are accessed using the following metrics: f1-score, recall, accuracy, and precision. These measures have the following definition.

A. Performance Metrics

1) Accuracy: The percentage of test cases that a technique successfully detects on a given test set is its accuracy. It is computed as follows in Eq. (8).

$$Accuracy = \frac{RN+RP}{RP+AP+RN+AN} \quad (8)$$

2) Precision: The proportion of all positively identified instances to the number of accurately detected positive occurrences by the model is known as precision. It is quantified as in Eq. (9).

$$Precision = \frac{True\ Positives}{(True\ Positives+False\ Positives)} \quad (9)$$

A value between 0 and 1, where 1 denotes perfect precision and 0 denotes no correct positive predictions, is the accuracy level.

3) Recall: The notion of the positive cases that the model properly identifies is known as recall. It is computed as follows in Eq. (10).

$$Recall(sensitivity) = \frac{True\ Positives}{True\ Positives+False\ Negatives} \quad (10)$$

4) F1-Score: The F1 score is a widely used statistic to evaluate how well sorting models perform in detection tasks; it is especially helpful for algorithms that function well in threat detection and prediction. The F1 score is useful when a dataset is uneven, meaning that one class significantly outnumbers the other. Equation is used to evaluate the F1 score as shown in Eq. (11).

$$F1\ Score = 2 \times \frac{(Precision*Recall)}{(Precision+Recall)} \quad (11)$$

One should take the F1 score into account when evaluating someone since it offers a helpful and impartial way to assess recall and accuracy. When choosing between accuracy and recall, as is frequently encountered in detection tasks, it is a useful metric to use.

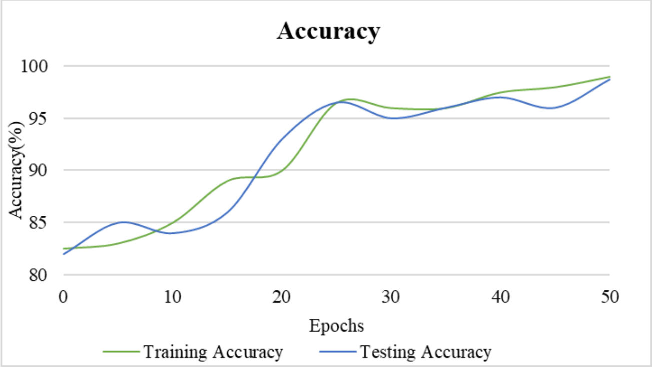

Fig. 3. Training and testing accuracy of the proposed AE-MLP approach.

The AE-MLP model's testing and training accuracy are shown in Fig. 3. The following graph shows how well the model works in two different phases training, when it learns from the data, and testing, when it applies newfound knowledge to previously unknown data. The model's resilience and dependability in real-world circumstances are indicated by the tight alignment of the training and testing accuracies, which implies that the model generalizes effectively to fresh data.

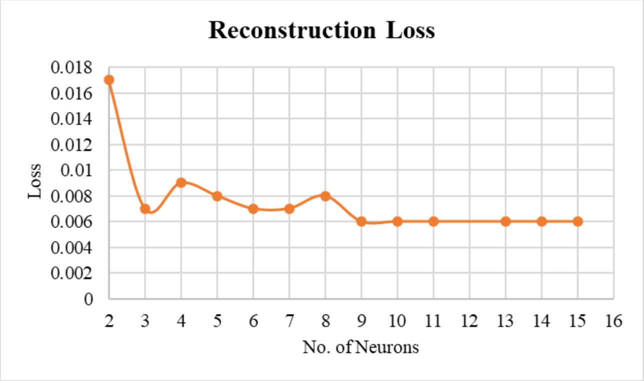

Fig. 4. Reconstruction loss of the proposed approach.

The reconstruction loss of the suggested method, as depicted in Fig. 4, serves as a critical indicator of the auto encoder model's proficiency in recovering its input data. A smaller reconstruction loss not only signifies the model's capability to capture and represent underlying data patterns accurately but also implies a higher fidelity in reconstructing normal network behaviour. This robust representation enables the AE-MLP approach to effectively discern anomalous activities, thereby bolstering its capacity for precise and reliable cyber threat detection.

B. *Consideration with Other State-of the-Art Approaches*

The need for threat detection in the contemporary cyber environment has led to much study on the subject. For such cases, researchers have used a variety of powerful and advanced ML techniques. This section compares the accuracy of our method against various cutting-edge detection algorithms based on traditional ML and DL approaches using the NSL-KDD dataset.

As seen in Table I, proposed auto encoder-MLP strategy have produced superior results than alternative approaches and it is depicted in Fig. 5. It compares the AE-MLP approach's accuracy (99%), precision (98.75%), recall (98.92%), and F1-score (98.79%) with alternative techniques. The proposed AE-MLP technique outperforms the conventional RNN (83.28%), STL+SVM (84.96%) and AE+DNN (94.21%) methods in terms of accuracy.

TABLE I. COMPARISON WITH EXISTING METHODS AND SUGGESTED METHOD

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1Score (%)
RNN [21]	83.28	81.60	82.24	81.42
STL+SVM [22]	84.96	81.78	84.08	80.21
AE+DNN [11]	94.21	92.78	90.82	90.21
Proposed AE+MLP	99	98.75	98.92	98.79

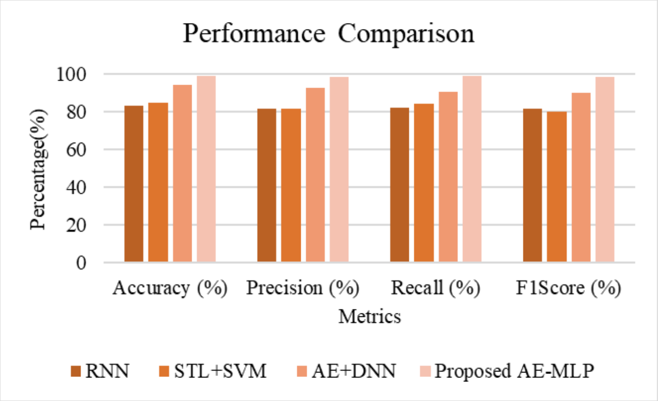

Fig. 5. The performance evaluations of AE-MLP with conventional approaches.

TABLE II. COMPARISON WITH DIFFERENT DATABASE WITH SUGGESTED METHOD

Different Dataset	Accuracy (%)
KDD99 [23]	98
UNSW-NB15 [24]	97
Proposed NSL-KDD	99

Table II shows the accuracy of the proposed technique on three different datasets: UNSW-NB15, KDD99, and the proposed NSL-KDD dataset. The approach demonstrated 98% accuracy on the KDD99 dataset, a reputable intrusion detection benchmark. The technique achieved 97% accuracy rate on another popular benchmark, the UNSW-NB15 dataset. Notably, it attained the greatest accuracy of 99% on the NSL-KDD dataset that was particularly created for the suggested technique. These findings highlight the method's efficacy in correctly categorising instances of network traffic and detecting intrusions; the NSL-KDD dataset shows especially impressive performance, perhaps because it aligns with the method's methods and methodologies.

TABLE III. DETECTION SPEED COMPARISON OF INTRUSION DETECTION METHODS

Approach	Detection Speed (seconds)
RNN [21]	100
STL+SVM [22]	120
AE+DNN [11]	90
Proposed AE+MLP	80

Table III presents a concise summary of the detection speeds of different intrusion detection techniques. It also includes a comparison of detection speeds of intrusion detection methods. While STL+SVM [22] shows a little slower speed of 120 seconds, RNN [21] indicates a detection speed of 100 seconds. Using autoencoders and deep neural networks, AE+DNN [11] provides a 90-second detection time quicker than previous methods.

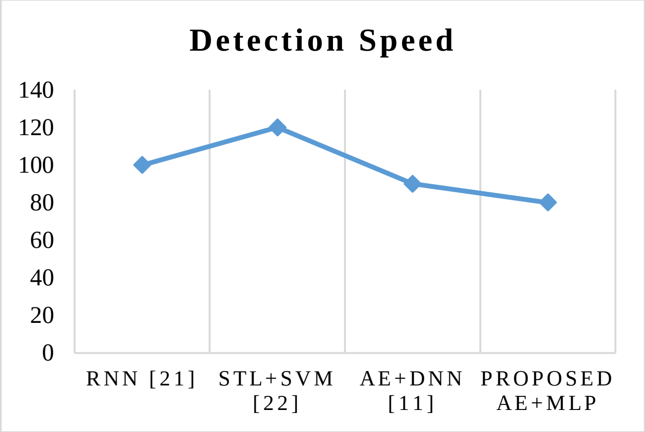

Fig. 6. The detection speed comparison evaluations of AE-MLP with conventional approaches.

The fastest detection speed, achieved by the suggested AE+MLP approach at 80 seconds, demonstrates the effectiveness of integrating autoencoders with multilayer perceptron's. The findings highlight how crucial it is to take into account detection speed when choosing an intrusion detection technique for cybersecurity applications, in addition to other performance parameters like accuracy and scalability. Fig. 6 illustrate the Detection Speed Comparison Evaluations of AE-MLP with Conventional Approaches.

A. *Discussion*

The examined works provide several methods for using machine learning (ML) and deep learning (DL) techniques in intrusion detection systems (IDS). In order to construct a generalised intrusion detection model, I. H. Sarker, Y. B. Abushark, F. Alsolami, and A. I. Khan [10] provide "IntruDTree," an ML-based framework that ranks security characteristics in order of importance and exhibits efficacy across a variety of cyber security datasets. Its dependence on fixed feature rankings and static datasets, however, could make it more difficult to respond to emerging threats. G. Muhammad, M. S. Hossain, and S. Garg [11] offer an intrusion detection system (IDS) that relies on learnt representations and achieves high accuracy but could be subject to adversarial assaults. The system uses stacked autoencoders and a deep neural network. In comparison to traditional models, M. Alsaedi, F. A. Ghaleb, F. Saeed, J. Ahmad, and M. Alasl [12] approach to improving dangerous URL identification is based on cyber threat information and involves a two-stage ensemble learning process. However, the performance of the system may be impacted by potential biases from internet searches. Utilising SVM and naive Bayes feature embedding, J. Gu and S. Lu [13] describe an effective IDS approach that shows excellent accuracy across a variety of datasets but runs into scaling issues as attack types diversify. J. Du, K. Yang, Y. Hu, and L. Jiang, [14] present NIDS-CNNLSTM, a computationally demanding intrusion detection and classification system for IIoT wireless sensing. These studies demonstrate the variety of methodologies employed for IDS creation, each with advantages and disadvantages in dealing with the always changing cyber threat environment.

The result section compares and assesses a novel cyber threat detection technique termed AE-MLP against existing cutting-edge approaches. With exceptional recall, accuracy, precision, and F1 scores of 99%, 98.75%, 98.92%, and 98.79%, respectively, the AE-MLP method stands out. These findings demonstrate its exceptional ability to correctly identify threats while reducing false alarms. In both the training and testing phases, when the model applies its newly acquired knowledge to previously unseen data, Fig. 3 offers a visual depiction of the model's performance. The model's resilience and dependability in real-world circumstances are indicated by the tight alignment of the training and testing accuracies, which implies that the model generalizes effectively to fresh data. The reconstruction loss of the suggested method is shown in Fig. 4. The auto encoder model's ability to recover its input data is indicated by its reconstruction loss. A smaller reconstruction loss suggests that the underlying patterns in the data can be captured and represented by the model with good accuracy. The figures provide empirical evidence supporting the efficacy of the AE-MLP approach and its potential to enhance cyber security measures in the face of evolving threats.

VI. Conclusion and Future Scope

In conclusion, the use of Auto Encoder-MLP hybrid models is a noteworthy development in the field of financial cyber security, providing a strong means of improving threat detection systems. This hybrid approach shows superior performance in identifying and mitigating potential threats within financial systems by combining the potent classification capabilities of MLP neural networks with the special powers of auto encoders to compress and reconstruct complex data representations. These models are highly skilled at identifying small irregularities that point to harmful behaviours like fraud, data breaches, and attempts at unauthorized access. This is achieved by the thorough examination of a variety of financial information, including transaction records, user activity patterns, and network traffic. The potential for further study and development in this area is bright. Hybrid model designs may be further improved and refined to increase their scalability and forecast accuracy, which would provide strong defense against advanced cyber-attacks. Furthermore, the contextual knowledge of cyber security risks might be enhanced by the integration of new data sources, such as social media feeds, market trends, and geopolitical indicators. This would allow for more thorough risk assessments and proactive threat mitigation techniques. Furthermore, developments in machine learning methods, especially in the areas of reinforcement learning and DL, present opportunities for enhancing the effectiveness and flexibility of financial cyber security systems over time.

References

[1] A. Dainotti, A. Pescapè, and G. Ventre, Worm Traffic Analysis and Characterization. 2007, p. 1442. doi: 10.1109/ICC.2007.241.

[2] N. Sun, J. Zhang, P. Rimba, S. Gao, L. Y. Zhang, and Y. Xiang, "Data-Driven Cybersecurity Incident Prediction: A Survey," IEEE Commun. Surv. Tutorials, vol. 21, no. 2, pp. 1744–1772, 2019, doi: 10.1109/COMST.2018.2885561.

[3] W. N. W. Manan and C. Y. Han, "Detection of Distributed Denial-of-Service (DDoS) Attack with Hyperparameter Tuning Based on Machine

Learning Approach," in 2023 7th International Symposium on Innovative Approaches in Smart Technologies (ISAS), Nov. 2023, pp. 1–8. doi: 10.1109/ISAS60782.2023.10391487.

[4] D. Ghelani, T. K. Hua, and S. K. R. Koduru, "Cyber Security Threats, Vulnerabilities, and Security Solutions Models in Banking," Preprints, preprint, Sep. 2022. doi: 10.22541/au.166385206.63311335/v1.

[5] S. Zeadally, E. Adi, Z. Baig, and I. A. Khan, "Harnessing Artificial Intelligence Capabilities to Improve Cybersecurity," IEEE Access, vol. 8, pp. 23817–23837, 2020, doi: 10.1109/ACCESS.2020.2968045.

[6] V. Koutsouvelis, S. Shiaeles, B. Ghita, and G. Bendiab, "Detection of Insider Threats using Artificial Intelligence and Visualisation," in 2020 6th IEEE Conference on Network Softwarization (NetSoft), Ghent, Belgium: IEEE, Jun. 2020, pp. 437–443. doi: 10.1109/NetSoft48620.2020.9165337.

[7] S. A. AlAjlan and A. K. J. Saudagar, "Machine learning approach for threat detection on social media posts containing Arabic text," Evol. Intel., vol. 14, no. 2, pp. 811–822, Jun. 2021, doi: 10.1007/s12065-020-00458-w.

[8] W. Hu and H. Hu, "Discriminant Deep Feature Learning based on joint supervision Loss and Multi-layer Feature Fusion for heterogeneous face recognition," Computer Vision and Image Understanding, vol. 184, pp. 9–21, Jul. 2019, doi: 10.1016/j.cviu.2019.04.003.

[9] P. Shettar, A. V. Kachavimath, M. M. Mulla, N. D. G, and G. Hanchinmani, "Intrusion Detection System using MLP and Chaotic Neural Networks," in 2021 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India: IEEE, Jan. 2021, pp. 1–4. doi: 10.1109/ICCCI50826.2021.9457024.

[10] I. H. Sarker, Y. B. Abushark, F. Alsolami, and A. I. Khan, "IntruDTree: A Machine Learning Based Cyber Security Intrusion Detection Model," Symmetry, vol. 12, no. 5, Art. no. 5, May 2020, doi: 10.3390/sym12050754.

[11] G. Muhammad, M. S. Hossain, and S. Garg, "Stacked Autoencoder-Based Intrusion Detection System to Combat Financial Fraudulent," IEEE Internet of Things Journal, vol. 10, no. 3, pp. 2071–2078, Feb. 2023, doi: 10.1109/JIOT.2020.3041184.

[12] M. Alsaedi, F. A. Ghaleb, F. Saeed, J. Ahmad, and M. Alasli, "Cyber Threat Intelligence-Based Malicious URL Detection Model Using Ensemble Learning," Sensors, vol. 22, no. 9, Art. no. 9, Jan. 2022, doi: 10.3390/s22093373.

[13] J. Gu and S. Lu, "An effective intrusion detection approach using SVM with naïve Bayes feature embedding," Computers & Security, vol. 103, p. 102158, Apr. 2021, doi: 10.1016/j.cose.2020.102158.

[14] J. Du, K. Yang, Y. Hu, and L. Jiang, "NIDS-CNNLSTM: Network Intrusion Detection Classification Model Based on Deep Learning," IEEE Access, vol. 11, pp. 24808–24821, 2023, doi: 10.1109/ACCESS.2023.3254915.

[15] A. Bécue, I. Praça, and J. Gama, "Artificial intelligence, cyber-threats and Industry 4.0: challenges and opportunities," Artif Intell Rev, vol. 54, no. 5, pp. 3849–3886, Jun. 2021, doi: 10.1007/s10462-020-09942-2.

[16] M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," IEEE Symposium. Computational Intelligence for Security and Defense Applications, CISDA, vol. 2, Jul. 2009, doi: 10.1109/CISDA.2009.5356528.

[17] S. Yu, J. Wang, J. Liu, R. Sun, S. Kuang, and L. Sun, "Rapid Prediction of Respiratory Motion Based on Bidirectional Gated Recurrent Unit Network," IEEE Access, vol. 8, pp. 49424–49435, 2020, doi: 10.1109/ACCESS.2020.2980002.

[18] A. Soleimani and S. E. Khadem, "Early fault detection of rotating machinery through chaotic vibration feature extraction of experimental data sets," Chaos, Solitons & Fractals, vol. 78, pp. 61–75, Sep. 2015, doi: 10.1016/j.chaos.2015.06.018.

[19] Wei Song, "A new deep auto-encoder using multiscale reconstruction errors and weight update correlation," Information Sciences, vol. 559, pp. 130–152, Jun. 2021, doi: 10.1016/j.ins.2021.01.064.

[20] M. A. Setitra, M. Fan, B. L. Y. Agbley, and Z. E. A. Bensalem, "Optimized MLP-CNN Model to Enhance Detecting DDoS Attacks in SDN Environment," Network, vol. 3, no. 4, Art. no. 4, Dec. 2023, doi: 10.3390/network3040024.

[21] C. Yin, Y. Zhu, J. Fei, and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," IEEE Access, vol. 5, pp. 21954–21961, 2017, doi: 10.1109/ACCESS.2017.2762418.

[22] M. Al-Qatf, Y. Lasheng, M. Al-Habib, and K. Al-Sabahi, "Deep Learning Approach Combining Sparse Autoencoder With SVM for Network Intrusion Detection," IEEE Access, vol. 6, pp. 52843–52856, 2018, doi: 10.1109/ACCESS.2018.2869577.

[23] P. Illy, G. Kaddoum, C. Miranda Moreira, K. Kaur, and S. Garg, "Securing Fog-to-Things Environment Using Intrusion Detection System Based On Ensemble Learning," in 2019 IEEE Wireless Communications and Networking Conference (WCNC), Apr. 2019, pp. 1–7. doi: 10.1109/WCNC.2019.8885534.

[24] C. Liang, B. Shanmugam, S. Azam, M. Jonkman, F. D. Boer, and G. Narayansamy, "Intrusion Detection System for Internet of Things based on a Machine Learning approach," in 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), Mar. 2019, pp. 1–6. doi: 10.1109/ViTECoN.2019.8899448.