Received: October 26, 2022. Revised: December 5, 2022. 35

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Hybrid AE-MLP: Hybrid Deep Learning Model Based on Autoencoder and Multilayer Perceptron Model for Intrusion Detection System

Duaa Al-Safaar[1]* **Wathiq Laftah Al-Yaseen[2]**

[1]*College of Science for Women, University of Babylon, Babylon, Iraq*
[2]*Kerbala Technical Institute, Al-Furat Al-Awsat Technical University, 56001, Kerbala, Iraq*
* Corresponding Author's Email: wathiq@atu.edu.iq

Abstract: Network violations are currently society's major challenge. For networks to be protected against hostile threats, an intrusion detection system (IDS) is important. To create effective IDS, deep learning (DL) is used in various fields, including information security. In this paper, a hybrid deep learning approach is proposed to effectively identify network intrusions using Autoencoder (AE) and Multi-layer perceptron (MLP). We use Autoencoder which can reduce the number of the original attributes based on the number of attributes, we first enter the original data on the Autoencoder and produce new compressed data, then enter it on the MLP classifier. The NSL-KDD dataset is thoroughly evaluated to determine the efficacy of the hybrid AE-MLP model the best outcomes are reached, with an accuracy rate of 87.6% and 81.06% (binary classification and multi-classification). In addition, the proposed hybrid method was compared with various recently proposed DL-based attack detection mechanisms. In terms of performance on the available dataset, it is observed that the proposed model outperformed.

Keywords: Intrusion detection system, Autoencoder, MLP, NSL-KDD, Deep learning.

1. Introduction

The Internet is becoming more and more important to our work and daily lives as Internet technology continues to improve and get better. However, a lot of data is created, processed, and transferred when people use and interact with the Internet. These data have been used for illegal purposes, posing serious risk protection of networks [1]. Fortunately, the intrusion detection system (IDS) [2] can effectively address such issues. IDS monitors networks or hosts as an active security technology and issues notifications when threats are found. Through the use of intrusion detection techniques, network attack behavior can be better protected by understanding it across simulation models and analyzing data. How to identify all types of unauthorized network activity, particularly unexpected hostility, is a critical problem that cannot be disregarded. There are two types of network activity: normal and malicious. Moreover, malicious can be divided into five different types of network traffic: normal, probing, denial of service (DoS), user to root (U2R), and root to local (R2L). Therefore, it is possible to consider intrusion detection as a problem of classifying [3]. The effectiveness of the classifier's performance in accurately recognizing harmful traffic can be improved, which would significantly increase the accuracy of intrusion detection. By enhancing the classifier's capability to accurately identify malicious traffic, IDS accuracy can be increased.

Deep learning (DL) approaches have lately acquired prominence as strong techniques Due to promising outcomes in the disciplines of image processing, computer vision (CV), natural language processing (NLP), and other areas [4]. Numerous deep-learning techniques have recently been used in intrusion detection. To achieve intrusion detection, deep learning techniques can automatically extract features and classify, such as autoencoder (AE), multi-layer perceptron (MLP), long short term memory (LSTM) and convolutional neural networks (CNN).

Through previous researches, it became clear to us that auto encoder gave good results in many fields such as medicine, text encoding, images processing, etc. This technique has recently been used in the field of intrusion detection producing impressive detection results.

In addition to that, the multi-layer perceptron (MLP) method has been adopted in most of the researches and there are also good results. MLP applications in diverse fields such as speech recognition, image recognition, and machine translation software. In the field of intrusion detection, this technique has produced outstanding detection outcomes.

C. zhang et al. [5], introduced a deep learning based approach for IDS that can be used to address the issue. The autoencoder is used in the proposed approach, and it was decided to use the encoder of deep autoencoder to press lower significant attributes and without the use of a decoder, extract crucial attributes. Using the indicated method, anybody can construct a network and recognize threats more quickly. The model is evaluated by using NSL-KDD datasets for 5 – class, the accuracy rate is 79.74%.

M. AL-QATF et al. [6], proposed a successful deep learning strategy built on the IDS for self-taught learning (STL). The sparse autoencoder (SAE) and SVM are combined in the suggested method to learn features and reduce dimensionality. It effectively increases the support vector machines (SVM) attack prediction accuracy while reducing the training and testing times by a significant amount. The model is evaluated by using the NSL-KDD dataset for binary and multi-classification, the accuracy rate was 84.96% and 80.48% respectively.

M. Yousefi-Azar et al. [7] proposed to learn the latent representation of various feature sets using techniques that utilize auto-encoders (AEs). They demonstrate how effectively the AE can learn a logical concept of semantic similarity between input data automatically. AE-Gaussian Naive Bayes. The model was evaluated by using the NSL-KDD dataset for binary classification, they achieved an accuracy rate of 83.34%.

S. Z. Lin1et al. [8], introduced 11 layers make up the CNN structure in char CNN-IDS, comprising 4 completely linked tiers and 7 convolutional tiers. The approach is evaluated by using NSL-KDD datasets for binary class and multi class, the accuracy rate is 85.07%, 79.05 respectively.

K. WU et al. [9], presented a CNN model to address the unbalanced data set drawback. This model would automatically detect traffic characteristics from unprocessed data sets and determine the weighting coefficient from every class's cost function depending on its numbers. The model increases the class's accuracy when dealing with small numbers while simultaneously lowering the false alarm rate (FAR). They changed the raw traffic vector format into image format to further lower calculating costs. Used the common NSL-KDD data set to evaluate how well the suggested approach performed and the accuracy of the Conv model is 79.48%.

M. Moukhafi* and K. El Yassini [10], a neural network-based suggested IDS approach based the multi-layer perceptron (MLP) and genetic algorithm (GA). Network traffic is classified into normal and harmful communications using an MLP classifier. The MLP classification model's architecture is improved using GA. The performance of the proposed method evaluated used the KDD99 dataset that was employed to train the GA-MLP methodology. This technique has accurate results up to 93.05%.

Through previous researches, it became clear to us that AE and MLP gave good results compared with other method such as CNN and LSTM and other methods of machine learning.

The effectiveness of intrusion detection still needs to be improved, therefore, we suggest a new approach to detecting network intrusions, AE-MLP: Hybrid Deep Learning Model based Autoencoder and Multilayer Perceptron model for Intrusion Detection System. A basic autoencoders objective is to rebuild the input data, which can reduce the number of the original features based on the number of features that we specify.

So, in the proposed method we first enter the original data on the Autoencoder and reduce the number of the original features based on the number of features that we specify. Then, the training phase was implemented on data and the new encoding data on training and testing data to predict the classification by using just the encoder phase (non-symmetric). This method called Nonsymmetric Autoencoder, which like autoencoder structure but utilizing just the encoder phase (non-symmetric) after make the training dataset. This entails the suggested change from the symmetric encoder-decoder architecture to relying solely on the encoder phase (Nonsymmetric).

This is justified by the fact that, given the right ability tests, it is possible to minimize the impact on accuracy and efficiency while reducing computational and time overheads, then enter it on the MLP classifier. We used MLP Classifier, which is characterized by choosing the best epochs and the best batch size. We worked to train and test for the data. The NSL-KDD dataset is fully evaluated to

determine the usefulness of the hybrid AE-MLP model, and the best performance is achieved, obtaining the highest accuracy 87.6% and 81.06% (binary class and multi-class).

The following are the paper's primary contributions and findings:

1. A hybrid deep learning approach, which combines autoencoder (AE) and MLP classifier is proposed. Which offers a fresh approach to intrusion detection also, IDS accuracy is increased by the proposed approach.
2. We present an AE model that encodes the original input features to obtain new features that are compressed and different from the original data, then this new data is sent to the MLP classifier.
3. To test our suggested system, we employ the NSL-KDD dataset. The empirical findings demonstrate that the hybrid AE-MLP model outperforms conventional approaches.

This paper is organized as follows: in section 1 the introduction about IDS and deep learning, in section 2 related works the existing state-of-the-art in improved detection and classification using a deep learning-based neural network, overview of deep learning methods implemented in this paper in section 3, the frame work of the proposed model and experiment results in section 4, and in the section 5 the conclusion and future works.

2. Related works

The most crucial component of the protection networking systems architecture for information security is intrusion detection. IDS uses a variety of machine learning and deep learning techniques to identify threats from normal network traffic. We review the existing state-of-the-art in improved detection and classification using a deep learning based neural network here. The summary of these related works is shown in Table 1.

C. zhang et al. [5], proposed a deep learning-based approach for IDS that can be used to address the issue. The autoencoder is used in the proposed approach, and it was decided to use the encoder of deep autoencoder to press lower significant attributes and without the use of a decoder, extract crucial attributes. Using the indicated method, anybody can construct a network and recognize attacks more quickly. The model is evaluated by using NSL-KDD datasets for 5 – class, the accuracy rate is 79.74%.

M. AL-QATF et al. [6], suggested a successful deep learning strategy built on the IDS for self-taught learning (STL). The sparse autoencoder (SAE) and SVM are combined in the suggested method to learn features and reduce dimensionality. It effectively increases the support vector machines (SVM) attack prediction accuracy while reducing the training and testing times by a significant amount. The model is evaluated by using the NSL-KDD dataset for binary and multi-classification, apply 10-fold cross-validation KDDTrain+ to evaluate the performance of model for five-category classification. The accuracy rate was 84.96% and 80.48% respectively.

M. Yousefi-Azar et al. [7], proposed to learn the latent representation of various feature sets using techniques that utilize auto-encoders (AEs). They demonstrate how effectively the AE can learn a logical concept of semantic similarity between input data automatically. AE-Gaussian Naive Bayes. The model was evaluated by using the NSL-KDD dataset for binary classification, they achieved an accuracy rate of 83.34%.

S. Z. Lin et al. [8], introduced 11 layers make up the CNN structure in char CNN-IDS, comprising 4 completely linked tiers and 7 convolutional tiers. The approach is evaluated by using NSL-KDD datasets for binary class and multi class, the accuracy rate is 85.07%, 79.05 respectively.

K. WU et al. [9], presented a CNN model to address the issue of an unbalanced data set. This model would automatically identify traffic features from raw data sets and determine the cost function weight coefficient of each class based on its numbers. The model increases the accuracy of the class with small numbers while simultaneously lowering the false alarm rate (FAR). They changed the raw traffic vector format into image format to further lower calculating costs. Used the common NSL-KDD data set to evaluate how well the suggested approach performed and the accuracy of the Conv model is 79.48%.

G. Kamdem et al. [11], proposed building a convolutional neural network that can identify breaches in a cyber-physical system by utilizing recent advancements in deep learning. A feature extraction technique with a cross-entropy loss function has been employed with CNN. There are 448 features in the output features based on the NSL-KDD training data. After that, A CNN classifier was used to categorize the 448 characteristics that were produced by the CNN features extraction using two separate class classifications. The NSL- KDD dataset is used to evaluate the proposed CNN-CNN intrusion detection scheme. The model is evaluated by using NSL-KDD datasets for binary class and multi-class, the accuracy rate is 80.07% and 77.15% respectively.

Y. Zhang et al. [12], proposed multi-layer auto-encoder architecture for training, each auto-encoder architecture is Tier following tier is trained so that the

Table 1. The summary of these related works

Reference	Description	Performance (accuracy)
C. zhang et al. [5]	The employment of an autoencoder allowed for the compression of less crucial characteristics and the extraction of crucial features without the need for a decoder.	Used NSL-KDD datasets Accuracy 79.74% in multi classification.
M. AL-QATF et al. [6]	Self-taught learning (STL), is used for feature learning and dimensionality reduction by collection of the sparse autoencoder (SAE) with SVM.	Used NSL-KDD dataset. The accuracy in 2 class 84.96% and in 5 class 80.48%.
M. Yousefi-Azar et al. [7]	AE-Gaussian Naive Bayes.	Used NSL-KDD dataset, the accuracy rate is 83.34% for 2 class
S. Z. Lin1 et al. [8]	Proposed CNN architecture in char CNN-IDS has 4 completely linked tiers and 7 convolutional tiers.	Used NSL-KDD datasets Accuracy rate is 85.07% in 2 class, 79.05% in 5-class.
K. WU et al. [9]	Used CNN to select traffic features and To address the issue of an unbalanced data set.	Used NSL-KDD datasets. Accuracy rate is 79.48%.
G. Kamdem et al. [11]	CNN used as features extraction method and then output from the CNN were pass to a CNN classifier to be classified.	Used NSL- KDD dataset. accuracy rate in 2 class 80.07%, in 5 class 77.15%
Y. Zhanga et al. [12]	Multi-layer auto-encoder network is used for training and the output AE is the input of the LSTM method is used to classification prediction for the data.	Used KDDcup99 datasets. The accuracy rate for 5 kinds of attack 97.6%, 96.8%, and 95.3%, 94.8% and 94.7% respectively.
B. A. TAMA et al. [13]	Hybrid feature selection strategy that involves 3 methods (particle swarm optimization, ant colony method, and genetic algorithm).	In NSL-KDD dataset, the accuracy rate is 85.8%, In UNSW-NB15 the accuracy is 91.27 %.In 2 class.
C. YIN et al. [14]	Recurrent Neural Networks (RNN-IDS).	used the NSL-KDD dataset 83.28 % in 2 class.
Yukun Wu 1, 2 et al. [15]	Stacked autoencoder (SAE) with an SVM and the kernel approximation technique.	Used NSL-KDD dataset the accuracy is 85.8% in 2 class.
Z. Li et al. [27]	Recurrent neural networks with a variable number of hidden layers: Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU).	Used NSL-KDD dataset 81.34 in 2 class

data size is progressively decreased, and the outcome of one AE becomes the input of the next. The LSTM method is used to classify the predicted data after dimension reduction. KDDcup99 was used, 10% of which were test and training sets. Used 10% KDDcup99 dataset. The accuracy rate of this technique for 5 different violence types was 97.6%, 96.8%, and 95.3%, 94.8%, and 94.7% respectively.

B. A. TAMA et al. [13], proposed enhanced IDS using two-level classifier ensembles and hybrid feature selection. To minimize the feature size of the training datasets, a hybrid feature selection strategy that combines 3 methods particle swarm optimization, ant colony algorithm, and genetic algorithm is used. NSL-KDD and UNSW-NB15 are discussed in this study. A reduced error pruning tree (REPT) classifier's classification performance is taken into account when choosing features. Then it is suggested to use a two-level classifier ensemble based on rotation forest and bagging, two meta-learners. On the NSL-KDD dataset. The suggested algorithm displays 85.8% accuracy and on the UNSW-NB15 the accuracy is 91, 27%.

C. YIN et al. [14], presented a deep learning approach for detecting intrusions based recurrent neural networks (RNN-IDS). Moreover, they studied the number of neurons and different learning rate that impacts the performance of the proposed model. The model is evaluated by using the NSL-KDD dataset for binary classification and they achieved an accuracy rate of 83.28 %.

Y. Wu 1, 2 et al. [15], proposed stacked autoencoder (SAE), a support vector machine (SVM), and the kernel approximation method are combined in a joint training model. It was able to utilize a linear SVM to uniformly approximate to the Gaussian kernel SVM thanks to the training model's usage of the SAE to achieve decrease of the attribute dimension, random Fourier attributes to accomplish kernel approximation, and explicit application of random Fourier translating to the sample's subset. The SAE then engages in cooperative training with the effective linear SVM. The NSL-KDD dataset was employed to verify the model and the accuracy for binary classification was 85.8%.

Z. Li et al. [27], proposed Recurrent neural networks with a variable number of hidden layers: long short-term memory (LSTM) and gated recurrent unit (GRU), also evaluate the recently proposed broad learning system (BLS) and its extensions. The models are trained and tested using border gateway protocol (BGP) datasets that contain routing records collected from reseaux IP Europeens (RIPE) and BCNET as well as the NLS-KDD dataset containing network connection records. The algorithms are compared based on accuracy and F-score. The accuracy for binary classification was 81.34 %.

AE demonstrates better performance than other traditional machine learning classification techniques. The above works show that most researchers try to combine various techniques to achieve high accuracy for IDS. AE achieves good results compared with other deep learning and machine learning techniques such as CNN and LSTM and SVM. So, there is still room for improvement in the performance of IDS. Therefore we believe the proposed work in this research can improve the current research domain.

3. Background

In this section, a brief overview of deep learning methods implemented in this paper is provided. It includes:

1. Autoencoders (AE) and their types.
2. Multi-layer perceptron (MLP).

3.1 Autoencoder (AE)

It is an artificial neural network that teaches a reduced (encoded) version of the input data. With an initial tier that is smaller neurons in their hidden tiers and the exact size as the previous tier, these networks have a unique hourglass shape. A basic autoencoders objective is to rebuild the input data so that $x_i'$is as similar to x_i as possible. To put it another way, an autoencoder picks up a rough estimate of the identity function. The problem is that the network must identify patterns and constructions in the input data in order to encode it (in a hidden tier) and subsequently decode it (output tier) because there are only a finite number of hidden neurons [16]. There are two symmetrical elements in an auto encoder: an encoder and a decoder. The encoder takes the raw data and derives features, whereas the decoder uses those characteristics to reconstruct the data. Slowly reducing the discrepancy between the decoder's output and the encoder's input occurs throughout training. The encoder's factors reflect the data's core if the decoder is able to recreate the data using the extracted features. The AE constructions includes an input tier, a latent tier, and an output tier. As shown in Figure 1 [17], the input tier and the output tier are equal in size, and the size of the latent tier must be smaller than that of the input tier.

For a given training dataset $x = \{x_1, x_2, \dots, x_m\}$ with m samples, where xi is a d-dimensional attribute vector, the encoder maps the input vector xi to a hidden representation vector h_i through a deterministic mapping f_Θ as given in Eq. (1)

$$h_i = f_\Theta(x_i) = s(w_{xi} + b) \quad (1)$$

Where w , $\acute{d} \times d$, $\acute{d}$ the number of hidden units, b bias vector, and the θ mapping parameter set $\theta = \{w, b\}$. s Sigmoid activation function denoted as

$$s(t) = \frac{1}{1+\exp - t} \quad (2)$$

Where parameter t determined the form of the function. The decoder converts return the resulting hidden representation h_i to a rebuilt d-dimensional vector y_i in input space as:

$$y_i = g_{\acute{\theta}(x_i)} = s\left(w_{h_i} + \acute{b}\right) \quad (3)$$

Where $\acute{w}$ is a $d \times \acute{d}$, $\acute{b}$ is a bias vector and $\acute{\theta} = \{\{\acute{w}, \acute{b}\}\}$. The autoencoder must be trained to reduce the variance between input and output. Consequently, the below equation is used to calculate a loss function:

$$\mathcal{L}(x, y) = {}^{1}/_{m} \sum m_i = 1 \, \|x_i - y_i\|2 \quad (4)$$

Where m the number of the training dataset. The important goal is to find the best parameters $(\Theta \; and \; \acute{\Theta})$. This efficiently reduces the disparity among input and rebuilt output across the whole training set as:

$$\Theta = \{w, b\} = arg_\Theta min L(x, y). \quad (5)$$

There are many types of autoencoder used for various purposes.

1. Stacked autoencoder

AEs are stacked one on top of the other to create SAE. The next AE is trained to utilize the latent tier from the initial AE after the single-tier AE has been trained. By doing this step again.

To obtain greedy hierarchical learning, AEs are tired, with the lth latent tier serving as input to the stack's l + 1st latent tier. Rather than initializing the weights at random, the SAE outcomes are utilized to

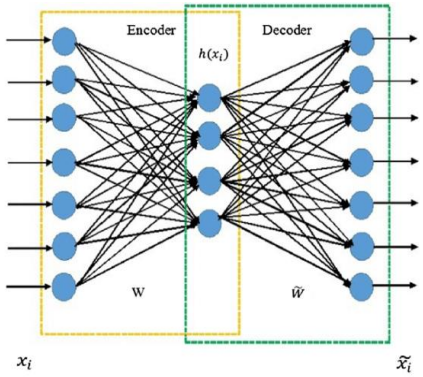

Figure. 1 The structure of AE

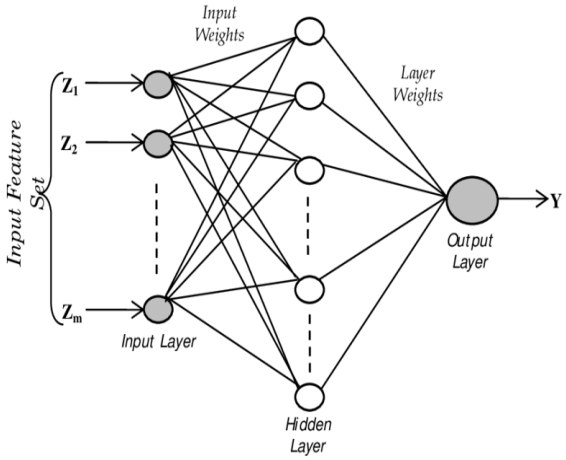

Figure. 2 The Structure of MLP

pre-train the fully linked DNN. This technique aids in improving the optimal effect and initializing model parameters around desirable local minimum values [16].

2. Sparse autoencoder

It is used for unsupervised reconstruction of new feature representation and dimension reduction. The sparse auto-encoder needs less training time and enhances the prediction accuracy [18].

3. Denoising autoencoder

It works by rebuilding the original input after part damaging the actual model input in order to train an autoencoder algorithm to provide a resilient input feature representation. [19].

4. Variational autoencoder

In contrast to conventional autoencoders, it requires the hidden representation's probability to adhere to a specified probability. In principle, variational autoencoders work similarly to stacked autoencoders in that the encoder creates a hidden representation, and the decoder reconstructs the input data using the hidden representation [19].

5. Nonsymmetric autoencoder

It has many non-symmetric hidden tiers and an autoencoder. Essentially, this entails the suggested change from the symmetric encoder-decoder architecture to relying solely on the encoder phase (Nonsymmetric). This is justified by the fact that, given the right ability tests, it is possible to minimize the impact on precision and efficiency while reducing computational and time overheads. NDAE functions effectively as a scalable hierarchical unsupervised feature extractor for high-dimensional inputs. [20].

3.2 Multi-layer perceptron (MLP)

It is one very common varieties of ANN, is an artificial neural network that feeds information forward. Consists of an input layer, one or more hidden layers, and an output layer. One or more neurons make up each layer, and their job is to apply what is known as an "activation function" to the input. The neuron also adds a value known as "bias," and each link between nodes has a weight corresponding to its significance in the network. Deep MLP is defined as having more than one hidden layer. MLP is used for solving problems that require supervised learning as well as research into computational neuroscience and parallel distributed processing. Applications include speech recognition, image recognition, and machine translation. MLPs were popularly employed for pattern identification and handwriting recognition in the 1980s. MLPs are frequently employed for intrusion detection since they are among the simplest neural network architectures to develop [21]. Fig. 2 illustrates the structure of the MLP model [22].

Numerous neurons make up each tier, and each neuron is completely coupled with the neuron in the tier below by nodes connected. As a result, the network can be expressed generally as:

$$Y = F\left[\sum_{j=1}^{m} W_{kj} \cdot F\left[\sum_{i=1}^{n} W_{ij} X_i + B_j\right] + B_k\right] \quad (5)$$

Where W_{ij} are the weights between hidden and output tier, W_{kj} are weights between input and hidden tiers; and X_i are input parameters; m is the amount of neurons in a hidden tier; n is the amount of neurons in an input tier, B_j and B_k are the bias values of the neurons in the hidden and output tiers, respectively; F is the transfer function; and Y is the output.

4. Hybrid AE-MLP proposed model

In this paper, a hybrid deep learning approach is proposed to effectively identify network intrusions using Autoencoder (AE) and Multi-layer perceptron

Table 2. The representation of instances in NSL-KDD dataset for training and testing.

Types of Attack	Number of instances	
	Training Dataset	Test Dataset
Normal	67343	9711
Denial of service (DoS)	45927	7456
Probe	11656	2421
Remote to local (R2L)	995	2756
User to root (U2R)	52	200
Total	125973	22544

(MLP). This proposed has three phases described which are:

Data collection and Pre-processing: including data cleaning, representation of data and normalization of data.

Classification algorithm: it is hybrid deep learning model based Autoencoder (AE) and MLP classifier to improve the efficiency of the IDS.

Two categories are utilized to evaluate performance:

• The sample labels are changed to a normal class and four broad classes of attacks, which are Probe, DoS, R2L, and U2R, in order to categorize multiple classes.
• The desired labels are changed into an "attack" and a "normal" class in order to categorize two classes.

4.1 Phase 1. Data collection and preprocessing

The popular benchmarks NSL-KDD datasets are used to evaluate the proposed methods in all phases. Pre-processing: including data cleaning, representation of data and normalization of data.

4.1.1. Datasets

A number of datasets have been produced to help in the evaluation and comparison process of intrusion detection systems. The NSL-KDD datasets are among these datasets that used widely for evaluating the performance of the IDS. Therefore, we will use these datasets to evaluate the performance of hybrid DL-IDS in all phases.

NSL-KDD dataset was enhanced version of KDDCup'99 intrusion detection benchmark dataset to solve some of the inherent problems. Table 3.1. Illustrate the Distribution of instances in NSL-KDD dataset for training and testing [23]. Each instance in the dataset displays 41 continuous and discrete attributes (38 numerical and 3 symbolic), to more details [24, 25]. Basic features, content features, traffic features, and Host-based features are the four categories of features in NSL-KDD.

As a result, NSL-KDD avoids the problem of categorization bias. Duplicate and superfluous records were also removed using the NSL-KDD, reducing the overall volume to a reasonable level. As a result, the tests may be run on the entire dataset, and the results from other studies are similar and consistent. To some extent, the NSL-KDD solves the concerns of information loss and bias.

1. Normal.
2. DoS (denial of service): An attacker tries to prevent legitimate users from using a network resource by consuming the bandwidth or by overloading computational resources.
3. Probe: An attacker tries to gain information about target system prior to initiating an attack.
4. R2L Remote to user. Sending a packet over a network to a target host, an attacker who does not have an account, uses various vulnerabilities to acquire entry as a user on that host.
5. U2R User to Root: If an attacker gets into the system, they begin out with a regular account. Then, they use system flaws to get root access to the system.

4.1.2. Preprocessing

The data preprocessing contains three stages: data cleaning, representation and normalization of data that discuss below.

1. Data cleaning

To assure the quality of the data, data cleaning is a crucial data mining operation that is typically carried out prior to model training. Although the NSL-KDD data set has been updated, it has been discovered that the value of the 20th feature, "num_outbound_cmds," is always 0. As a result, this feature is meaningless and should be eliminated. Each record in the data collection has 40 features after data cleaning.

2. Representation of data

In NSL-KDD dataset every record has 41 features, containing 7 symbolic features and 34 continuous features. To easy process this features, symbolic features should be converted to numeric features. [26] We used the one hot encoding method to encoded the features "protocol_type", ''service'', and ''flag'', which contain 3, 70, and 11 characters, respectively. As example the feature "protocol_type" contains three types of characters ''tcp'', ''udp'' and ''icmp'', and their numeric values are ''100'', ''010'', and ''001''. After numeralization each record contains 121 features, the features ''land'', ''logged_in'', ''is_hot_login'', and ''is_guest_login''

include just ‘yes’ and ‘no’ characters, which can be simply encoded as 1 or 0.

3. Normalization of data

To minimize the effect of the numerical range of various features on model training. In this study, the value is scaled to [0, 1] via normalization, and the normalized value x' is

$$\acute{X} = (X - X_{min})/(X_{max} - X_{min}) \quad (6)$$

Where X is the starting value, $X_{max}\ and\ X_{min}$ are the features heights and lowest values. Additionally, logarithmic normalization would be a better choice for the features ‘‘duration’’, ‘‘src_bytes’’, and ‘‘dst_bytes’’, which have a greater range of values.

We make two normalization in preprocessing of data phase, one after representation of data and the second after autoencoder model.

4.2 Phase 2. Classification algorithm

Recently, deep learning has been high efficiency in the construction of intrusion detection systems. A hybrid model is proposed based Auto Encoder and MLP that produce improvement of IDS performance. This proposed model can achieve high efficiency in classification the network data into normal and abnormal with high accuracy, Detection rates and false alarm rates. We use the whole 10% KDD training and testing datasets to evaluate this model and we compare the proposed method with the other methods from deep learning that used the same datasets. The framework of the proposed hybrid deep learning AE-MLP model is shown in Fig. 3.

We provide a new approach to network intrusion detection, AE-MLP: Hybrid deep learning model based on autoencoder and multilayer perceptron model for intrusion detection system. Using the Autoencoder, A basic autoencoders objective is to rebuild the input data, which can reduce the number of the original features based on the number of features that we specify.

So, in the proposed method we first enter the original data on the autoencoder and reduce the number of the original features based on the number of features that we specify. Then, the training phase was implemented on data and the new encoding data on training and testing data to predict the classification by using just the encoder phase (non-symmetric). This method called nonsymmetric autoencoder, which like autoencoder structure but utilizing just the encoder phase (non-symmetric) after make the training dataset. This entails the suggested change from the symmetric encoder-decoder architecture to relying solely on the encoder phase (Nonsymmetric). This is justified by the fact that, given the right ability tests, it is possible to minimize the impact on accuracy and efficiency while reducing computational and time overheads, then enter it on the MLP classifier. We used MLP Classifier, which is characterized by choosing the best epochs and the best batch size. We worked to train and test for the data. We make two normalization in preprocessing of data phase, one after representation of data and the second after autoencoder model.

A new normalization preprocess should be applied on the resulting dataset from the previous step to become suitable to MLP Classifier. The NSL-KDD dataset is thoroughly evaluated to determine the efficacy of the hybrid AE-MLP model, and the best outcomes are obtained, with the prediction accuracy reaching 87.6% and%81.06 (binary-class and multi-class). The trained AE-MLP classifier is then used to input the test samples in order to identify violent acts. Steps for the hybrid AE-MLP model are shown in Fig. 4.

4.3 Phase 3. Experimental outcomes and evaluation

4.3.1. Experimental setting

We implement examination to determine the effectiveness of the proposed hybrid AE-MLP. Our proposed was implemented in the python / ANACONDA / Spyder environment version 3.9.12, with processor 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz 2.42 GHz and 8.00 GB RAM, 64-bit operating system, x64-based processor and Windows 10 Pro.

4.3.2. Hyper parameters of the proposed model

We implement experiments on the NSL-KDD dataset to evaluate the efficiency of the proposed model and examine the outcomes of the proposed hybrid AE-MLP with common deep learning methods. Tables 3 and 4 illustrate the parameters in binary and multi-classification in this method.

In multi-classification, the parameters of AE are: in the encoder, Dense 1 is 128 and Dense 2 is 64. In the decoder, Dense 1 is 64 and Dense 2 is 128, with epochs 10 and batch size 500. The optimizer of AE is Adam, and the activation function of each tier is ReLU. And the activation function of the output tier is softmax. The parameters of MLP classifier is [random_state = 1, learning_rate_init = 0.08, alpha = 0.004, epsilon = 1e-4].

Table 3. Training parameters in binary classification on NSL-KDD

Algorithms	Hyper parameters	Values
AE	Encoder \	
	Dense 1	200
	Dense 2	64
	Decoder \	
	Dense1	64
	Dense 2	200
	Epochs	10
	Batch size	500
	The optimizer	Adam
	Activation function of each layer	RELU
	Activation function of output layer	Softmax
MLP	Random state	1
	Learning rate	0.7
	Alpha	0.008
	Epsilon	Le-8

Table 4. Training parameters in multi classification on NSL-KDD

Algorithms	Hyper parameters	Values
AE	Encoder \	
	Dense 1	128
	Dense 2	64
	Decoder \	
	Dense1	64
	Dense 2	128
	Epochs	10
	Batch size	500
	The optimizer	Adam
	Activation function of each layer	RELU
	Activation function of output layer	Softmax
MLP	Random state	1
	Learning rate	0.08
	Alpha	0.004
	Epsilon	Le-4

Table 5. Confusion matrix.

Total population		Predicted condition	
		Normal	Anomaly
Actual condition	Normal	TN	FP
	Anomaly	FN	TP

In binary classification, the parameters of AE are: in the encoder, Dense 1 is 128 and Dense 2 is 64. In the decoder, Dense 1 is 64 and Dense 2 is 128 with epochs 10 and batch size 200. The optimizer of AE is Adam, and the activation function of each tier is ReLU. And the activation function of the output tier is softmax. The parameters of MLP classifier is [random_state = 1, learning_rate_init = 0.7, alpha = 0.008, epsilon = 1e-8].

During the training stage, each of these parameters is learned by trial and error.

4.3.3. Performance metrics

Accuracy, or the percentage of correctly classified data, is a widely used performance evaluation measure. Table 2.4 provides an example of the confusion matrix using the terms true positive (TP), true negative (TN), false positive (FP), and false negative (FN).

The accuracy is as follows:

$$\text{Accuracy} = (TP + TN)/ (TP + TN + FP + FN) \quad (7)$$

The following variables are also used to measure the model's performance because of the significant variation in record counts between categories in the NSL-KDD data set [26].

$$\text{Recall} = TP/ (TP + FN) \quad (8)$$

$$FPR = FP/ (FP + TN) \quad (9)$$

$$\text{Precision} = TP/ (TP + FP) \quad (10)$$

$$F - \text{score} = 2TP/ (2TP + FN + FP) \quad (11)$$

Where the recall is the percentage of accurately identified positives, also known as the true positive rate (TPR) or detection rate (DR); The percentage of negatives that are wrongly forecasted as positives is known as the false positive rate (FPR), often referred to as the false alarm rate (FAR); The f-score is the harmonic average of recall and precision, where precision is the percentage of expected to actual positives.

4.3.4. Results and discussion

Depends on the NSL-KDD dataset, two sets of experiments have been created to determine the efficiency of the hybrid AE-MLP model in binary class and multi-class, respectively. Based on criteria like accuracy, precision, f-value and detection rate, we evaluate the model's performance. Lastly, based on the most recent study, we compare the efficiency of hybrid AE-MLP model with some other methods.

The confusion matrix generated by the hybrid AE-MLP model on the NSL-KDD dataset is shown in Fig. 5 and Fig. 6. Fig. 5 and Fig. 6 represent the experimental results of the hybrid AE-MLP model forthe 2-class and 5-class classification, respectively. The experimental results show that most samples are

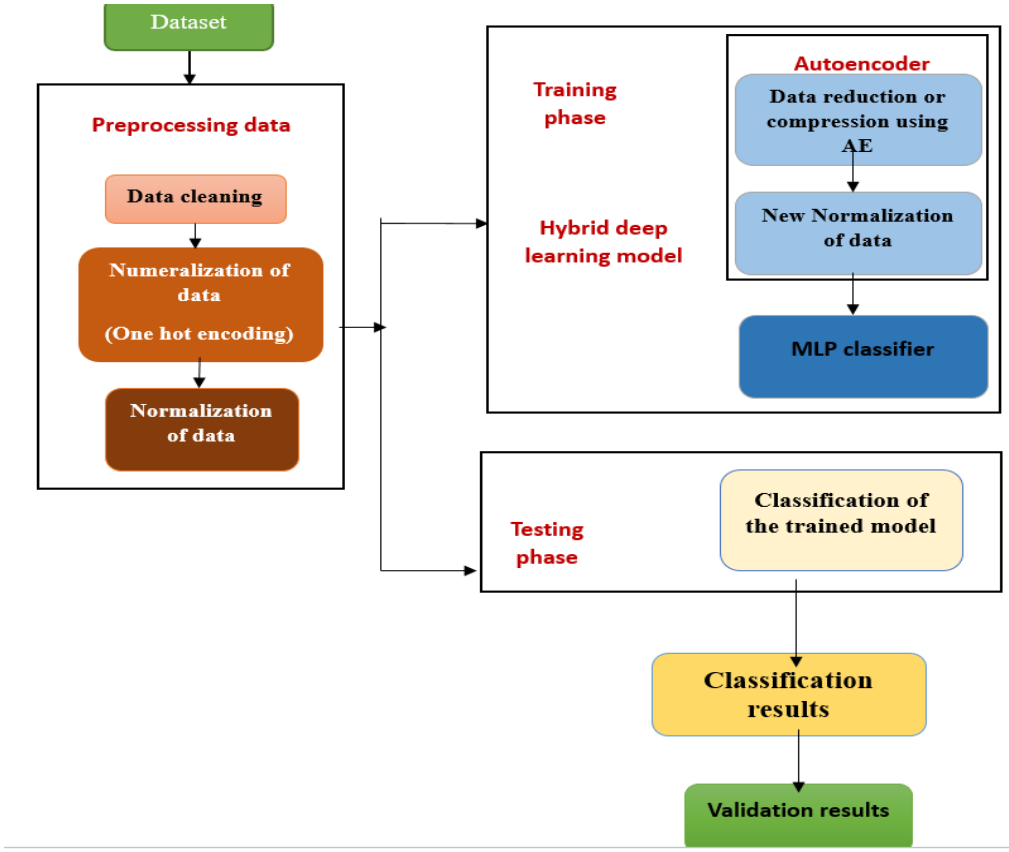

Figure. 3 The framework of the proposed Hybrid AE-MLP model for IDS

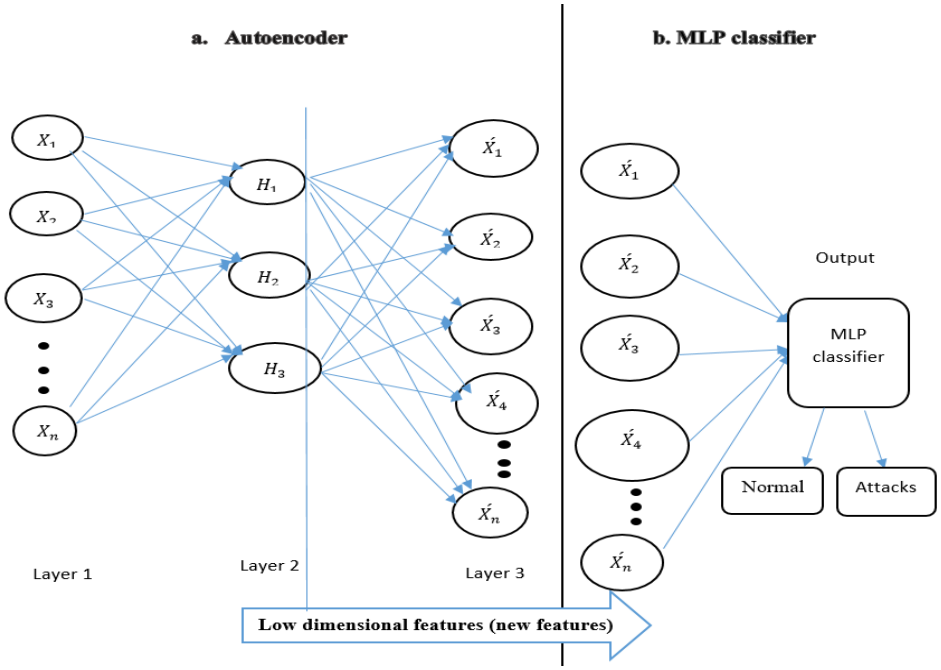

Figure. 4 Hybrid deep learning AE – MLP model steps

concentrated on the diagonal of the confusion matrix, indicating that the overall classification performance is very high. However, it can be intuitively seen the confusion matrix in Fig. 5 shows that the hybrid AE-MLP model achieves good detection performance in distinguishing normal traffics from attack traffic.

Table 6 show the evaluation results of hybrid AE-MLP model evaluate the binary-classification and multi-classification results from four evaluation indexes of accuracy, precision, recall, and F1-score. We can see that in the binary-classifications, the

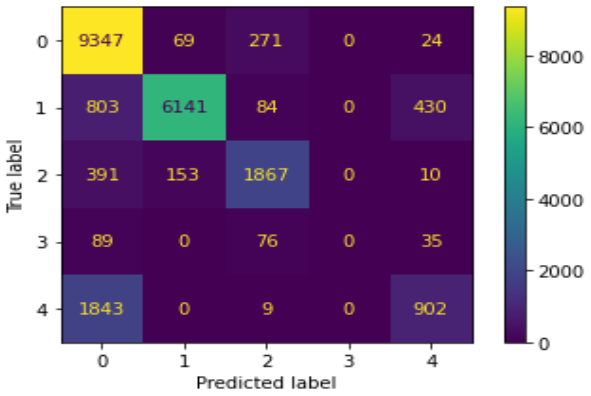

Figure. 5 Confusion matrix for multi classification

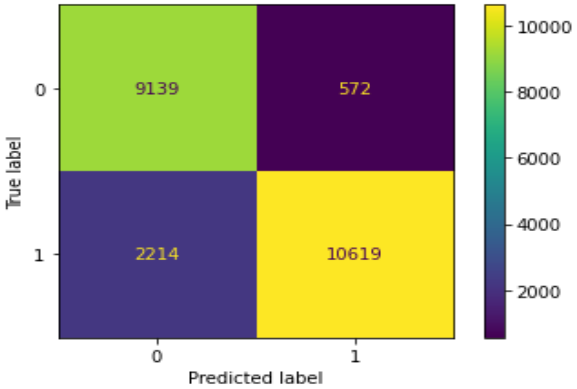

Figure. 6 Confusion matrix for binary classification

accuracy of proposed model is 87.6% and the precision, recall rate, and F1-score are 90.80, 0.94, and 0.87 respectively. In the multi-classification results, we can also see results the accuracy of the proposed model is 81.06% and the precision, recall rate, and F1-score are 0.75, 0.96, and 0.84 respectively.

Table 7 shows the comparison of the evaluation results of the hybrid AE-MLP model, AE, SAE-SVM, CNN, CNN-CNN, and RNN, on NSL-KDDTest sets, and evaluates the binary-classification and multi-classification results from four evaluation indexes of accuracy, precision, recall, and F1-score. We can see that the accuracy of the hybrid AE-MLP model is higher than the other methods respectively. Similarly, the precision, recall rate, and F1 score are higher, respectively. From this group of comparisons, we can see that the proposed model can effectively improve the accuracy of intrusion detection.

Fig. 7 shows the comparison evaluation results of the proposed model hybrid AE-MLP, AE, MLP, CNN, SAE-SVM, CNN-CNN, RNN, and AE-Gaussian Naïve Bayes on NSL-KDD datasets, and shows evaluate results of the binary classification and multi-classification from the evaluation of accuracy.

	Precision	recall	f1-score	support
0	0.75	0.96	0.84	9711
1	0.97	0.82	0.89	7458
2	0.81	0.77	0.79	2421
3	0.00	0.00	0.00	200
4	0.64	0.33	0.43	2754
Accuracy			**0.81**	**22544**
Macro avg	0.63	0.58	0.59	22544
Weighted avg	0.81	0.81	0.79	22544

	Precision	recall	f1-score	support
0	0.80	0.94	0.87	9711
1	0.95	0.83	0.88	12833
Accuracy			**0.88**	**22544**
Macro avg	0.88	0.88	0.88	22544
Weighted avg	0.89	0.88	0.88	22544

4 Conclusion and future work

In this paper, we propose an intrusion detection method, hybrid deep learning AE-MLP model in the proposed method we first enter the original data on the Autoencoder and reduce the number of the original features based on the number of features that we specify. Then, the training phase was implemented on data and the new encoding data on training and testing data to predict the classification by using just the encoder phase (non-symmetric). This method called Nonsymmetric Autoencoder, which like autoencoder structure but utilizing just the encoder phase (non-symmetric) after make the training dataset. This entails the suggested change from the symmetric encoder-decoder architecture to relying solely on the encoder phase (Nonsymmetric). This is justified by the fact that, given the right ability tests, it is possible to minimize the impact on accuracy and efficiency while reducing computational and time overheads, then enter it on the MLP classifier. We used MLP Classifier, which is characterized by choosing the best epochs and the best batch size. We worked to train and test for the data.

The NSL-KDD dataset is fully evaluated to determine the usefulness of the hybrid AE-MLP model, and the best performance is achieved, obtaining the highest accuracy 87.6% and 81.06% (binary class and multi-class). The intrusion detection performance of hybrid AE-MLP model is evaluated on the NSL-KDD dataset and the hybrid AE-MLP model the proposal made in this paper has a superior effect on intrusion detection. When compared to current classifiers (such as CNN, RNN,

Table 6. Measures in 2 class and 5 class on NSL-KDD datasets

Classification	Category	Accuracy	Precision (%)	Detection rate (%)	F-value (%)
Multi- Classification	Normal	96.52%	0.75	0.96	0.84
	DoS	82.34%	0.97	0.82	0.89
	U2R	77.11%	0.81	0.77	0.79
	R2L	0.00	0.00	0.00	0.00
	Probe	32.75%	0.64	0.33	0.43
	Overall	**81.06%**	0.75	0.96	0.84
Binary Classification	Normal	94.10%	0.80	0.94	0.87
	Attack	82.74%	0.95	0.83	0.88
	Overall	**87.6 %**	0.80	0.94	0.87

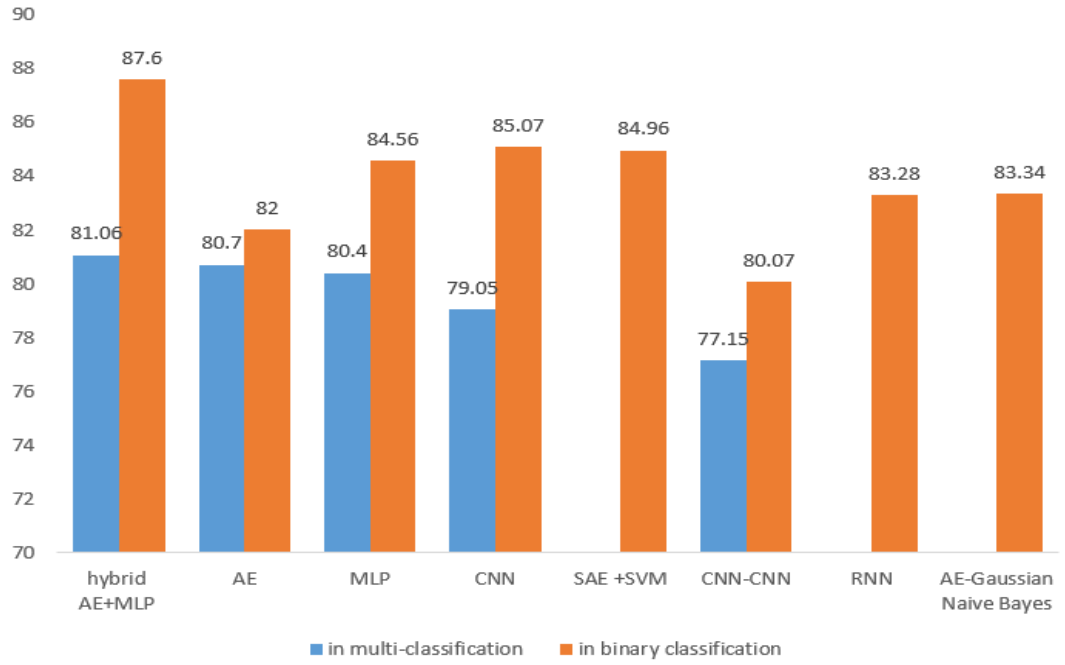

Figure. 7 The proposed method comparison with other methods in NSL-KDD dataset

AE, AE-LSTM, and SAE-SVM), also has higher accuracy.

. Reference studies, as well as this research, focused on the importance of intrusion detection systems in protecting information from attacks sophisticated and growing spread across the Internet and various types of networks.

• The reference studies included different algorithms and approaches that were applied and tested in the design of intrusion detection systems research machine learning techniques in designing an intrusion detection system, then the results were compared with the results of the techniques in previous studies.

• This research aims to improve the detection rates of the two and four types of attacks within the data set the NSL-KDD obtained in previous research improved the false alarm rates to be as low as possible.

Table 7. The Proposed method comparison with other methods in NSL-KDD dataset

Papers	**Techniques**	**Dataset**	**Accuracy**	**Precision (%)**	**Recall (%)**	**F-value (%)**
C. zhang et al. [5]	AE	Used NSL-KDD datasets (125973 training, 22544 testing dataset).	79.74% in 5-class	0.82	0.79	0.81
M. AL-QATF et al. [6]	SAE + SVM	Used NSL-KDD dataset (Used 10-cross validation).	In 2 class 84.96%	96.23	76.56	85.28
M. Yousefi-Azar et al. [7]	AE-Gaussian Naive Bayes	Used NSL-KDD datasets (125973 training, 22544 testing dataset).	In 2 class 83.34%	-	-	-
S. Z. Lin1 et al. [8]	CNN	Used NSL-KDD datasets (125973 training, 22544 testing dataset).	79.05 in multi 85.07% in binary	91.68	81.12	86.08
G. Kamdem De Teyou et al. [11]	CNN-CNN	Used NSL-KDD datasets (125973 training, 22544 testing dataset).	In 2 class 80.07% In 5 class 77.15%	80.07 75.15	85.00 74.14	-
B. ADHI TAMA et al. [13]	Particle swarm optimization, ant colony algorithm, and genetic algorithm for feature selection, used a two-level classifier ensemble based on rotation forest and bagging.	Considered 20% of dataset, KDDTrain+, consists of 25,192 samples, with 13,499 anomalous and 11,743 normal samples. In addition, take into account two separated test sets, KDDTest+ (22,544 samples) and KDDTest21 (11,850 samples).	85.8% in binary classification	85.00	-	-
C. YIN et al. [14]	RNN	Used NSL-KDD dataset (used 10 cross validation).	83.28 % in 2 class.	-	-	-
Y. Wu 1,2 et al. [15]	(JSAE-FSVM) Autoencoder (SAE) + SVM + the kernel approximation technique.	Used NSL-KDD datasets (125973 training, 22544 testing dataset).	85.8% in 2 class 83.5% in 5 class	0.919 0.823	0.785 0.801	0.847 0.784
Z. Li et al. [27]	Recurrent neural networks with a variable number of hidden layers: Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU).	Used NSL-KDD datasets (125973 training, 22544 testing dataset).	81.34 in 2 class	-	-	81.99
Our proposed	**AE + MLP classifier**	Used NSL-KDD datasets (125973 training, 22544 testing dataset).	**81.06% in 5 class** **87.6 % in 2 class**	**0.80** **0.95**	**0.94** **0.83**	**0.87** **0.88**

The above works show that most researchers try to combine various techniques to achieve high accuracy for IDS; there is still room for improvement. Therefore we believe the proposed work in this research can improve the current research domain.

We aim to research a more practical technique to raise IDS performance in subsequent work. We hope to research more AE types. Such as variational autoencoder to improve the accuracy of detecting.

Author contributions

Conceptualization, D. Al-Safaar, and W.L. Al-Yaseen; methodology, D. Al-Safaar, and W.L. Al-Yaseen; software, D. Al-Safaar, and W.L. Al-Yaseen; validation, W.L. Al-Yaseen; formal analysis, W.L. Al-Yaseen; investigation, D. Al-Safaar, and W.L. Al-Yaseen; resources, D. Al-Safaar; data curation, D. Al-Safaar, and W.L. Al-Yaseen; writing—original draft preparation, D. Al-Safaar; writing—review and editing, D. Al-Safaar, and W.L. Al-Yaseen; visualization, D. Al-Safaar; supervision, W.L. Al-Yaseen; project administration, W.L. Al-Yaseen; funding acquisition, D. Al-Safaar, and W.L. Al-Yaseen.

Conflicts of interest

The authors declare no conflict of interest.

References

[1] C. Liu, Y. Liu, Y. Yan and J. Wang, "An Intrusion Detection Model With Hierarchical Attention Mechanism", *IEEE Access*, Vol. 8, pp. 67542–67554, 2020.

[2] S. Dwivedi, M. Vardhan and S. Tripathi, A. K. Shukla, "Implementation of adaptive scheme in evolutionary technique for anomaly-based intrusion detection", *Springer-Verlag GmbH Germany, Part of Springer Nature 2019*, Vol. 13, pp. 103–117, 2020.

[3] T. Su, H. Sun, J. Zhu, S. Wang and Y. Li, "BAT: Deep Learning Methods on Network Intrusion Detection Using NSL-KDD Dataset", *IEEE Access*, Vol. 8, pp. 29575–29585, 2020.

[4] O. Avci, O. Abdeljaber, S. Kiranyaz, M. Hussein, M. Gabbouj and D. J. Inman, "A review of vibration-based damage detection in civil structures: From traditional methods to Machine Learning and Deep Learning applications", *Mechanical Systems and Signal Processing*, Vol. 147, No. 15 January 2021.

[5] C. Zhang, F. Ruan, L. Yin, X. Chen, L. Zhai, and F. Liu, "A Deep Learning Approach For Network Intrusion Detection based on NSL-KDD Dataset", *IEEE 13th International Conference on Anti-counterfeiting, Security, and Identification (ASID)*, Xiamen, China, pp. 25-27 October 2019.

[6] M. A. Qatf, Y. Lasheng, M. A. habib, and K. A. Sabahi, "Deep Learning Approach Combining Sparse Autoencoder With SVM for Network Intrusion Detection", *IEEE Access*, Vol. 6, pp. 2169-3536, 2018.

[7] M. Y. Azar, V. Varadharajan, L. Hamey and U. Tupakula, "Autoencoder-based Feature Learning for Cyber Security Applications", *Faculty of Science and Engineering Macquarie University*, Sydney, NSW, Australia, 2017.

[8] S. Z. Lin, Y. Shi and Z. Xue, "Character-level Intrusion Detection Based on Convolutional Neural Networks", *International Joint Conference on Neural Networks (IJCNN)*, Rio de Janeiro, Brazil, pp. 8-13, July 2018.

[9] K. Wu, Z. Chen, and W. Li, "A Novel Intrusion Detection Model for a Massive Network Using Convolutional Neural Networks", *IEEE Access*, pp. 2169-3536, 2018.

[10] M. Moukhafi, K. E. Yassini and S. Bri, "Intelligent intrusion detection system using multilayer perceptron optimised by genetic algorithm", *International Journal of Computational Intelligence Studies*, Vol. 9, No. 3, 2020.

[11] G. Kamdem, D. Teyou, and J. Ziazet, "Convolutional Neural Network for Intrusion Detection System In Cyber Physical Systems", arXiv: 1905.03168, 2019.

[12] Y. Zhang, Y. Zhang, N. Zhang, and M. Xiaoa, "A network intrusion detection method based on deep learning with higher accuracy", *International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI2019)*, 2020.

[13] B. A. Tama, M. Comuzzi, and K. Rhee, "TSE-IDS: A Two-Stage Classifier Ensemble for Intelligent Anomaly-Based Intrusion Detection System", *IEEE Access*, pp. 2169-3536, 2019.

[14] C. Yin, Y. Zhu, J. Fei, and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks", *IEEE Access*, pp. 2169-3536, 2017.

[15] Y. Wu, W. W. Lee, X. Gong and H. Wang, "A Hybrid Intrusion Detection Model Combining SAE with Kernel Approximation in Internet of Things", *Mdpi Sensors*, Vol. 20, p. 5710, doi:10.3390/s20195710, 2020.

[16] F. Farahnakian and J. Heikkonen, "A Deep Auto-Encoder based Approach for Intrusion Detection System", *International Conference on Advanced Communications Technology (ICACT)*, Turku, Finland, February, pp. 11-14, 2018.

[17] C. Tang, N. Luktarhan, and Y. Zhao, "SAAE-DNN: Deep Learning Method on Intrusion Detection", *Symmetry*, Vol. 12, p. 1695, doi: 10.3390/sym12101695, 2020.

[18] S. W. Lee, H. M. sidqi, M. Mohammadi, S. Rashidi, A. M. Rahmani, M. Masdari, and M.

Hosseinzadeh, "Towards secure intrusion detection systems using deep learning techniques: Comprehensive analysis and review", *Journal of Network and Computer Applications*, Vol. 187, No. 1 August 2021.

[19] H. Choi, M. Kim, G. Lee, and W. Kim, "Unsupervised learning approach for network intrusion detection system using autoencoders", *The Journal Of Supercomputing*, Vol. 75, pp. 5597–5621, 2019.

[20] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A Deep Learning Approach to Network Intrusion Detection", *IEEE Transactions on Emerging Topics in Computational Intelligence*, November, 2017.

[21] M. Labonne, "Anomaly-based network intrusion detection using machine learning", *Cryptography and Security [cs.CR], Institut Polytechnique de Paris*, 2020.

[22] M. Moukhafi, K. E. Yassini, S. Bri, and K. Oufaska, "Intelligent intrusion detection system using multilayer perceptron optimized by genetic algorithm", *International Conference on Advanced Intelligent Systems for Sustainable Development*, pp 393–404, Vol. 7 March 2019.

[23] V. Pai, A. N. D. Devidas, "Comparative analysis of Machine Learning algorithms for Intrusion Detection", *IOP Conf. Series: Materials Science and Engineering, Udupi, Karnataka*, India, Vol. 1013, 2021.

[24] S. W. Lin, K. C. Ying, C. Y. Lee, and Z. J. Lee, "Intelligent Algorithm with Feature Selection and Decision Rules Applied to Anomaly Intrusion Detection", *Applied Soft Computing Journal*, Vol. 12, No. 10, pp. 3285–3290, 2012.

[25] P. Somwang and W. Lilakiatsakun, "Anomaly Traffic Detection Based on PCA and SFAM", *The International Arab Journal of Information Technology*, Vol. 12, No. 3, May 2015.

[26] H. Hou, Y. Xu, M. Chen, Z. Liu, W. Guo, M. Gao, Y. Xin, and L. Cui, "Hierarchical Long Short-Term Memory Network for Cyberattack Detection", *IEEE Access*, 2020.

[27] Z. Li, A. L. G. Rios, G. Xu, and L. Trajkovic, "Machine Learning Techniques for Classifying Network Anomalies and Intrusions", *Natural Sciences and Engineering Research Council (NSERC) of Canada under Grant R31-611284*, 2019.