

---

# A Deep Learning Approach for Efficient Anomaly Detection in WSNs

Arul Jothi S, Venkatesan R

## Arul Jothi S\*

Department of Computer Science and Engineering  
PSG College of Technology, Coimbatore, 641004, India  
\*Corresponding author: [saj.cse@psgtech.ac.in](mailto:saj.cse@psgtech.ac.in)

## Venkatesan R

Department of Computer Science and Engineering  
PSG College of Technology, Coimbatore, India  
[rve.cse@psgtech.ac.in](mailto:rve.cse@psgtech.ac.in)

## Abstract

Data reliability in Wireless Sensor Networks (WSNs) has a substantial influence on their smooth functioning and resource limitations. In a WSN, the data aggregated from clustered sensor nodes are forwarded to the base station for analysis. Anomaly Detection (AD) focuses on detecting outlier data to ensure consistency during data aggregation. As WSNs have critical resource limitations concerning energy consumption and sensor node lifetime, AD is supposed to provide data integrity with minimum energy consumption, which has been an active research problem. Hence, researchers are striving for methods to improve the accuracy of data handled with a concern on the constraints of WSNs. This paper introduces a Feed-forward Autoencoder Neural Network (FANN) model to detect abnormal instances with improved accuracy and reduced energy consumption. The proposed model also acts as a False Positive Reducer intending to reduce false alarms. It has been compared with the other dominant unsupervised algorithms over robustness and other significant metrics with real-time datasets. Relatively, our proposed model yields an improved accuracy with fewer false alarms thereby supporting a sustainable WSN.

**Keywords:** Anomaly Detection, Autoencoder Neural Network, Data Aggregation, False Positive, Unsupervised Algorithms, Wireless Sensor Networks.

## 1 Introduction

Sensor networks make use of devices that detect or measure a physical property and can record, indicate, or respond to it. WSN assists observations and creates supremacy over physical environments from inaccessible locations with higher correctness. They are suitable to use in the fields such as natural observation, and armed forces to obtain sensing information. Sensor nodes used in a WSN have acute energy and cost limitations as they have to constantly gather and send data from the environment. This has restricted WSNs to follow critical resource constraints enabling sustainability in network deployment. In a WSN, sensor nodes are employed in a distributed manner in such a way that data gathering would continue even if some of them are dead due to power depletion or other

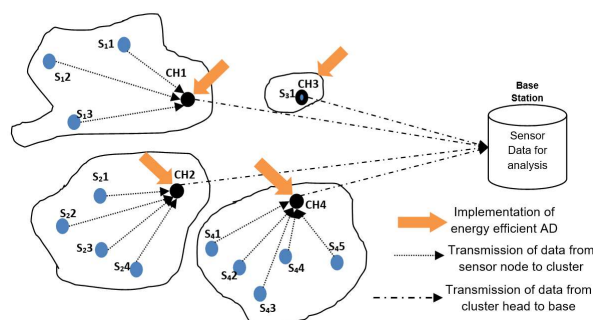


Figure 1: Distributed deployment scenario of WSN with anomaly detection in CHs

events. For continuous observance in the environment, data aggregation could be employed by forming clusters of sensor nodes with neighborhood nodes. Figure 1 depicts the implementation setup of a typical WSN that performs data aggregation in Cluster Heads (CHs). Clusters play a major role in increasing the lifetime of a WSN by the way of energy conservation, as all the data gathered by each sensor node are not blindly transmitted. The CH in each cluster is responsible for data aggregation and forwarding. The data aggregation process must ensure integrity by detecting and removing anomalous instances received from sensor nodes in a cluster [1]. These aggregated data after anomaly detection are forwarded to the base station suitable for further investigations and decisions.

### 1.1 Data aggregation in WSNs

Data aggregation is the method of scrutinizing the gathered data from several sensors, estimating the quantified response about the sensed environment, and providing fused information to the base station [2]. Data aggregation desires a scheme for converting the sensed data into high-quality information [3]. The major functions of Data aggregation are: avoiding data redundancy, reducing data transmission, and improving data accuracy [4, 5]. This can be achieved through AD during data aggregation. During data aggregation, the collection of data occurrences that are detected as erratic during aggregation amongst the residual of the dataset is defined as an anomaly or outlier [6]. Moreover, anomalies in sensor data are quantified as those important deviations in sensed data from the normal values [7]. Paying attention to anomalies and detecting abnormalities in data will increase data accuracy in WSNs [8]. The energy utilized for AD is a challenging task in a WSN due to its limitations on node lifetime. Deep learning provides solutions to overcome these issues with minimized resource utilization and extending the life expectancy of the network [9]. Our proposed work aims to perform AD during data aggregation in CH and reduce communication costs with reduced energy consumption.

### 1.2 Anomaly Detection in WSN

Sensing data are collected in the form of data streams in huge volumes of real observations from the environment and are stored in a database. Modern WSNs are aimed to collect multivariate types of data from the field instantaneously. Nowadays, AD models should be capable of dealing with data without any prior knowledge. This can be approached with unsupervised techniques that try to extract useful features from the dataset. Figure 1.1 depicts the implementation of the AD model in a single cluster which can be considered for all CHs in the network.

## 2 Literature Background

Researchers have focused on classical methods of unsupervised anomaly detection, which can be coarsely classified as clustering-based algorithms, classification algorithms, and deep learning methods.

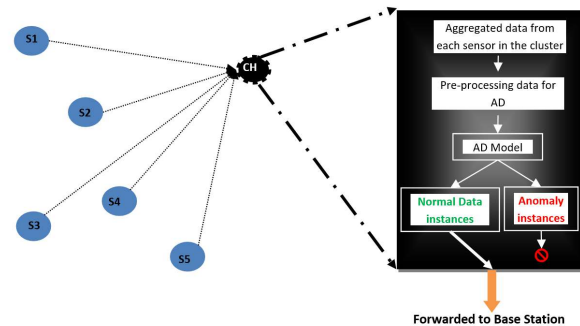


Figure 2: Implementation of AD model in a single cluster

Authors in [10] have implemented a prototype-based agglomerative hierarchical clustering method to identify features of uncertain data based on the probability distribution to be clustered and achieve average accuracy on all datasets. Ensemble clustering along with Gaussian Mixture Model and One-Class Support Vector Machine (OCSVM) [11] extracts features based on appropriate choice rather than manually by clustering anomalies into a specific type. Also, in [12] a hybrid combination of subspace clustering with OCSVM distinguishes between abnormal and normal instances based on mapping the features in the space as clusters. The classification-based approach used in [13] identifies outliers based on spatiotemporal correlation with Support Vector Data Description (SVDD). Moreover, in [14], fixing the neighborhood makes the algorithm converge faster and provides robustness with prior assumptions on several anomalies. The authors of [15] introduced OCSVM to be suitable for both univariate and multivariate datasets by training the model using histogram-based labeling. The subspace method, Robust Principal Component Analysis (RPCA) used in [16] performed low-rank and sparse regularization for Spatio-temporal data distribution and detection of the anomalies. rPCA has not considered the detailed information of Spatio-temporal data distribution. Many authors have used deep learning approaches [17] as it acts more suitable for unsupervised anomaly detection. In [18], a deep variational autoencoder with a feed-forward neural network model reduces inefficient learning in a semi-supervised context (for training) restraining the learning model in taking widespread values from the weight space.

Authors in [19] discuss the various deep learning techniques like Convolution Neural Network, Long Short-Term Memory, and Autoencoder which learn commonalities within the data to facilitate outlier detection. Hybrid models using deep learning techniques extract robust features within hidden layers. Distributed anomaly detection using an autoencoder neural network [20] model adapts the unforeseeable and new changes in a non-stationary environment. This model introduced a priority scheme to lower the false-positive rate. In [21], fast outlier detection was introduced by using Deep Belief Networks (DBN) to extract invariant features for complex and dimensional datasets.

Anomaly Detection using One-Class Neural Network in [22] discerns anomalies in complex datasets when the decision boundary between normal and anomalous is highly non-linear. Unsupervised anomaly detection algorithms in [23] aid a comparative evaluation of nearest neighbor-based algorithms, clustering-based algorithms, and statistical approaches dealing with both local and global anomalies. In recent years, Q-learning algorithms [24, 25] were implemented for handling reward noise and inverting gradient procedures for reducing loss. In addition, it is identified that deep reinforcement learning reduces the false-positive rates involving humans and learns meta-policy to optimize anomalies. A combination of the Gaussian process and Graph-based detection [26] with Hampel identifier overcomes the limitation of datasets having dense and sparse dimensions. This model identifies outliers in data across time. The fuzzy logical model in [27] uses neighborhood information with a fuzzy decision process to detect anomalies enabling a high detection rate and low false positive alarm rate. In [28], the authors have suggested recurrent neural networks with long short-term memory units (RNN with LSTM) to efficiently identify anomalies in health log data that was compiled from various

devices. The trials only used a tiny amount of data. In both instances, the data also needed to go through a quick preparation step before analysis. The data were organized to match each method's input. The model suggested in [29] operates in two phases and is effective in detecting network anomalies. Improved grey wolf optimization (ImGWO) is used for feature selection in the first phase to achieve the best possible trade-off between two goals, namely decreased error rate and feature-set minimization. In the second stage for network anomaly categorization, the model uses an improved Convolution Neural Network (ImCNN). The model suffers from high computation because of intricate design and parameter optimization during feature selection. To recognize outliers with a resourceful focus on the production of false alarms and incorrect conclusions, [30] created a method in which each cluster head was applied a denoising auto-encoder with a Gaussian kernel for outlier detection after performing a basic cluster algorithm based on the residual energies of the sensor nodes.

The literature review of related works indicates that the following research gaps (RG) exist in anomaly detection, which is worthwhile to ponder upon:

RG (i). The works in [14, 18, 23] indicate the need for tuning learning parameters to improve the model's detection. Additional activation functions and alternative structures are required to gauge the method's success [28].

RG (ii). The experiments in [19, 21, 22] express less detection accuracy with a low Area Under Curve (AUC) score.

RG (iii). In [11, 13, 14, 20, 22, 24, 29, 30] the models require high computational time that consumes energy when dealing with large datasets.

RG (iv). Subsequently, the methods in [12, 15, 19] suffer from a False Positive Rate (FPR).

RG (v). The methods in [23, 25, 27] perform poorly for high-dimensional data.

The proposed Feed-forward Autoencoder Neural Network (FANN) model is built to detect abnormal instances by employing multiple layers in the encoder and decoder to address a few of the aforementioned research issues. In particular, the focus is given to lowering the false positives, increasing the AD accuracy, and conserving the energy of the WSN without affecting the computational time.

### 3 Proposed Feed-Forward Autoencoder Neural Network for Anomaly Detection

#### 3.1 Overview

Autoencoders are typical neural networks that take input data, learn its significant characteristics, and produce output that is analogous to the input. Furthermore, the autoencoder learns to estimate a characteristics function, to construct an output value. The block diagram of the proposed model is shown in Figure 3, where initially normal instances are considered for training, which is archived in the encoding phase as regular data patterns through a series of hidden layers. The decoder in the network tries to reconstruct the data which is the same as the input normal pattern. When abnormal instances are fed, the model gathers the relevant regular instances in memory for reconstruction and generates an output. For abstract explanation, the archive is implemented as a single memory entry [31]. An additional component in the autoencoder model is included to compute Anomaly Score (AS) to reduce false alarms.

To ensure the durability of the model, the hyper-parameters like the number of layers for the encoder, the number of layers for the decoder, activation functions, and loss functions are tuned manually to get the appropriate threshold value for classification. In the proposed work, the autoencoder model tries to detect anomalies with fewer false positives, which is explained in the following section.

#### 3.2 Schema of FANN model

The proposed Feed-forward Autoencoder Neural Network (FANN) is composed of multiple hidden layers for encoding and decoding. Figure 3.1 shows the schema of it with an input layer and 6 dense layers encompassing 3 layers for encoding and 3 layers for decoding.

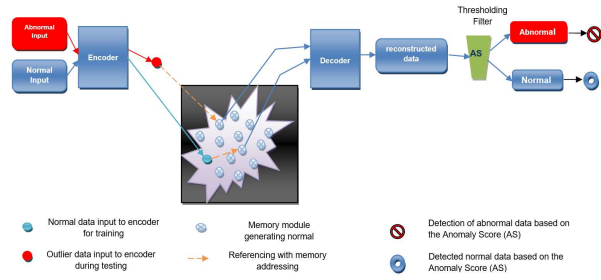


Figure 3: Block diagram of proposed Feed-forward Autoencoder Neural Network

### 3.3 Model Functionality

As portrayed in Figure 3.1 there are six dense layers indicated as f1, f2, f3, f4, f5, and f6, respectively. The fusion of all these tasks is represented as  $f(x) = f6(f5(f4(f3(f2(f1(x))))))$ . Generally, this process describes the structure of deep learning neural network. The procedure for unsupervised anomaly detection is discussed below:

1. In the preparation phase, the encoder in the proposed FANN model learns the given unlabeled data  $X_a$  by compressing data representation with the function  $f'$  and determines the attribute vector  $h_a$  from  $X_a$ :  
 $h_a = f'(X_a)$
2. The attribute vector  $h_a$  is considered for archiving the normal behavior of data instances. This memory archive data is used during the testing phase for evaluating the reconstruction error of the model.
3. The decoder tries to reproduce  $X_a$ : from the attribute vector  $h_a$  by a mapping function  $g'$ :  
 $r = g'(h_a) = g'(f'(X_a))$
4. Typically, training data instances cannot be used to estimate the characteristics of hidden layers but it encompasses a hidden correlation with them. The learning phase in the model evolves a procedure to assess the characteristics of hidden layers to obtain the best possible outcomes at the output layer. Moreover, this learning feature of the model must trace the correlation in the hidden layers with training data instances.
5. In the testing phase, the learning algorithm tries to interpret the knowledge gained during training with the same procedure of encoding and decoding to reproduce data instances. The reconstruction error of the model is generated using the loss function (discussed in section 3.4) to analyze the model's detection accuracy.
6. Each dense layer's weights are initialized randomly using Xavier initialization [32] such that the variance across each layer is the same. Equation 1 shows the random process initialization applied to all weights and equation 2 is used for estimating the variance of weights. Xavier initialization is considered efficient for activation functions since it prevents gradients from diminishing or setting off. For each layer Weight is calculated as

$$W_{(m,n)}^{[l]} = N\left(0, \frac{1}{n^{[l-1]}}\right) \tag{1}$$

and variance V is calculated as

$$Var(W) = \frac{1}{n^{[l-1]}} \tag{2}$$

7. Moreover, the model needs to be employed with a suitable activation function and loss function for improving detection accuracy and reducing false positives. A comparative analysis of various activation functions and loss functions is presented in the proceeding sections.

### 3.4 Activation function analysis for the model

The activation function is considered one of the hyper-parameters that plays a significant role in the deep learning model. Choosing the best activation function in hidden layers and the output layer would enhance the model’s capability to do the desired task (In the proposed work context, the objective is to detect anomalous data). In [33, 34] authors have discussed various activation functions suitable for deep learning techniques. Based on the study from existing works, the following activation functions with their respective equation mentioned in table 1 are considered for performance evaluation. The performance analysis is presented in section 4.3.

Table 1: Existing activation functions with its estimation formula

Reference	Activation Function	Equation
[34]	Hyperbolic Tangent	$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
	Rectified Linear Unit	$f(x) = \max(0, x) = \begin{cases} x_i, & \text{if } x_i \geq 0 \\ 0, & \text{if } x_i < 0 \end{cases}$
	Exponential Linear Unit	$f(x) = \begin{cases} \alpha(\exp(x) - 1) & \text{if } x \leq 0 \\ x & \text{for } x > 0 \end{cases}$
	Leaky ReLU	$f(x) = (ax + x) = \begin{cases} x & \text{for } x < 0 \\ ax & \text{for } x \geq 0 \end{cases}$
	Softmax	$f(x_i) = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$
	Swish	$f(x) = x.\text{sigmoid}(x) = \frac{x}{1 + e^{-x}}$
[35]	Gaussian Error Linear Unit	$f(x) = f(x) = xP(X \leq x) = x\Phi = x\frac{1}{2}[1 + \text{erf}(\frac{x}{\sqrt{2}})]$

### 3.5 Loss function analysis for the model

For training the proposed FANN with improved accuracy, reconstruction error must be reduced. Thus, two different loss functions are considered and their effectiveness is analyzed for model suitability. This section discusses the significance of Arctan Mean Square Error (AMSE) [6] and Huber Loss (HL) [36] functions which are considered for analysis.

#### 3.5.1 Arctan Mean Square Error (AMSE)

The Mean Square Error function (MSE) is an essential function that provides robust behavior of the model during the testing and training phase to identify data overfit when validation error goes beyond the training error. MSE of a model relating to test data can be estimated as the mean of the squared prediction errors overall data in the test set. Thus, the loss function of MSE represented in equation 3 calculates the dissimilarity between the true value and the predicted value for the data instance.

$$MSE = \frac{\sum_{(i=1)}^n (y_i - \lambda(x_i))^2}{n} \tag{3}$$

where,

$y_i$  is actual output value for test data instance  $x_i$ ,

$\lambda(x_i)$  is the reconstructed output value for test data instance  $x_i$  and

$n$  is the number of test data instances.

MSE is sensitive toward outliers, but minimizing MSE will respond better in identifying outliers effectively. Moreover, decreasing squared error will improve the accuracy of a model when the number of training data instances is known. Thus, a modified mean squared error value for training the model called Arctan MSE (AMSE) in equation 4 is considered. AMSE as discussed in [6] will gradually drop

the rate of loss for independent training data.

$$AMSE = \frac{1}{N} \sum_{(p=1)^p} \sum_{(i=1)^N} (\tan^{-1}((X_{pi} - (g(y))_{pi})))^2 \quad (4)$$

where,

$X_{pi}$  is the predicted value for data point  $i$ ,

$g(y)_{pi}$  is the actual value for the data point  $i$  and

$N$  is total number of data points

### 3.5.2 Huber Loss (HL)

Huber loss is more robust to outliers in data than the MSE loss which is suitable for differentiation by 0. There is a direct impact by tuning the hyper parameter,  $\delta$  (delta) to reduce the error. Thus Huber loss considers  $\delta = 0$  and  $\delta = \infty$  as like MSE, MAE and is given as:

$$L_{\delta}(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2, & |y - f(x)| \leq \delta \\ \delta * (|y - f(x)| - \frac{1}{2}\delta), & otherwise \end{cases} \quad (5)$$

where,

$\delta(y - f(x))^2$  is the difference between observed value and predicted value,

$\delta$  is the hyper-parameter delta, tuned to obtain minimum error (delta value chosen to converge to best solution is 0.1). Particularly, the loss reaches its minima by decreasing the gradient for MSE. Hence to avoid this, the HL function in equation 5 bends the minima in turn by decreasing the gradient by merging the optimum features from both MSE and MAE. For the proposed model, loss estimation is carried out by using HL and AMSE for 100 epochs. The analysis is presented in section 4.4.2.

## 3.6 False Positive rate reduction

Generally, anomaly detection may produce false positives which could affect the model's efficiency. To mitigate this drawback, the proposed FANN model is trained to perform precise prediction of actual anomalies by eliminating false positives with high recall and high precision. During the training phase, an acceptable threshold level is maintained for detecting abnormal data and false-positive rates. This will be considered over the test set also. An additional component (Thresholding Filter) is included in the model that removes false positives similar to [37]. To filter a false abnormal detection in the model, the results of the following were considered:

- Receiver Operator Characteristic (ROC) – ROC is capable of analyzing the model's performance based on various thresholds. The reason for choosing the ROC approach for false-positive reduction can be observed in section 4.2.
- Loss function – The outcome of reconstruction error from the loss functions viz, AMSE, HL is considered as the threshold to be fixed for filtering false positives in the model.
- Conformance threshold - As proposed in [38], the conformance threshold is evaluated based on the mean distribution of input instances. In this research, an assumption is made that anomalies detected in the model have more false positives if it reaches a threshold of more than 1%. For estimating the threshold value, the Central Limit Theorem (CLT) in probability theory is used. CLT performs sampling of the given dataset to a normal distribution. The threshold is calculated as:

$$Threshold(T) = (\mu(\# \text{ of anomalies}) - \sigma(\# \text{ of anomalies})) \quad (6)$$

where,

$\mu$  is the mean or average of anomalies and

$\sigma$  is the standard deviation of the anomalies

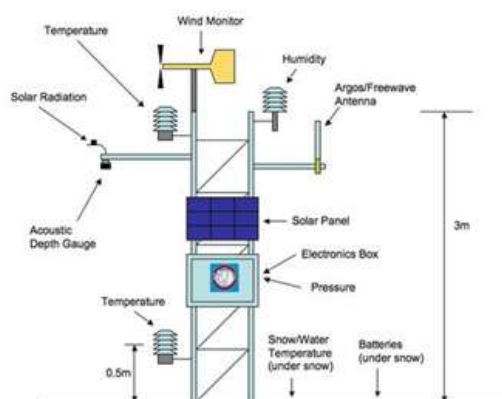


Figure 5: Sensors for weather observations in AWS

The Conformance threshold stated in equation 6 helps in maintaining a stable F1 score. This conformance threshold factor also impacts the performance of the model in terms of increasing the detection accuracy. The major intention of this research is to come up with a consistent model capable of categorizing abnormal instances.

## 4 Experimental Results

### 4.1 Dataset exploration

The performance of the proposed model is evaluated considering climate data [39]. Prediction of weather is considered to be the most important task in real-time. Many sensor stations are built around the world by various research labs to monitor climatic changes like temperature, humidity, pressure, precipitation, etc., These data need to ensure integrity for analysis like weather prediction, global warming rate, and disaster alerts related to the environment. Weather Underground is a service provider for real-time weather information retrieved with the help of Automatic Weather Stations (AWS) available to over 250,000 around the world. An AWS is a set of weather measuring sensors that can be installed either in-home or in business buildings. Figure 3.6 shows the sensor setup of a Weather Station (WS) utilized for weather observations. Indira Gandhi International Airport (IGIA) station acts as WS and provides real-time weather observation for DELHI weather forecasts. Airport Ws routinely issue reports used by pilots, air traffic controllers, meteorologists, climatologists, and other researchers.

In the Delhi region, 12 Automatic Weather Stations around a 50 km radius have been set up that measure 24x7 hours of weather in IGI airport. Here, ‘g’ aggregator nodes are considered in the WS network. A total of 20 features related to temperature, humidity, pressure, dew point, and wind speed with statistical readings are observed along with precipitation at each time interval. Nearly 20 years of climatic information are collected for AD investigation. These weather data are prone to errors, outages, and other defects. The following section discusses the implementation setup of AD using the FANN model.

### 4.2 Data processing and visualization

The dataset is considered as unlabeled pre-processed data with temporary labels assigned for each column. The normal instances and abnormal instances are categorized for generating a training set. Moreover, the test set data is generated with both normal instances and abnormal instances. The distribution of data is visualized in Figure 4.1. Box Plot representation provides a great benefit for understanding the outlier rate in the dataset. It is also observed that the dataset has a highly imbalanced dataset.



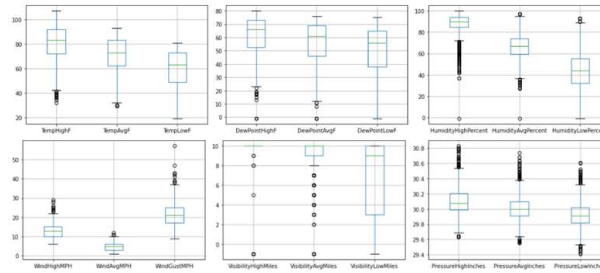


Figure 6: Data distribution analysis using Box plot

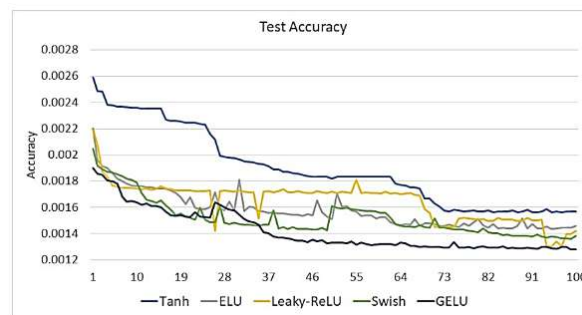


Figure 7: Accuracy of activation functions for 100 epochs

### 4.3 Analysis of activation functions with a loss function

The processed data is fed into the proposed FANN for training. As discussed in section 3.4, various activation functions have been experimented with for the given dataset. The result shown in Figure 4.2 depicts the line plot of the accuracy of the activation functions trained for 100 epochs. From the graph Figure 4.2 of activation function accuracy, Hyperbolic Tangent activation maintains a stable accuracy of data when the model is executed for 100 epochs. In many research works, Leaky-ReLU is considered a suitable activation function but it will affect the computation and may overfit the model losing linearity. Based on the proposed FANN model’s focus, Hyperbolic Tangent (Tanh) activation function is identified as appropriate for the model to detect anomalies with high accuracy and minimum error.

### 4.4 Methods for mitigating False alarm rate

#### 4.4.1 AUC - ROC Curve

To rate the performance of the model, the classified outcomes are measured using Receiver Operating Characteristics Curve repetitively. The Area Under the Curve (AUC), the integral of the ROC, is the probability that the classifier ranks based on positive examples for differentiating between positive and negative classes. This perceptive ensures that the AUC assessment method is more suitable for false-positive reduction and detection accuracy. For empirical illustration in unsupervised AD, AUC-based assessment is rationally considered to be the actual standard for estimation.

In Figure 4.4.1 AUC-ROC curve of the proposed FANN model provides the relationship between the true positive rate and the false positive rate for assessing detection accuracy. It is observed that

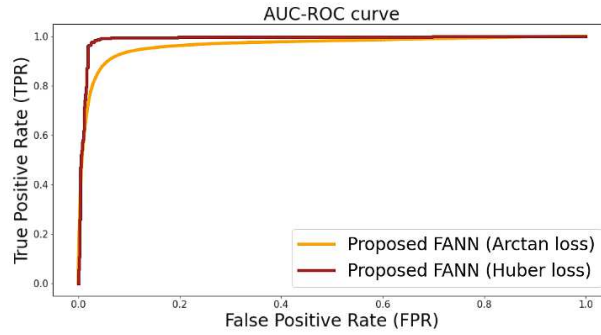


Figure 8: AUC-ROC curve of proposed FANN model

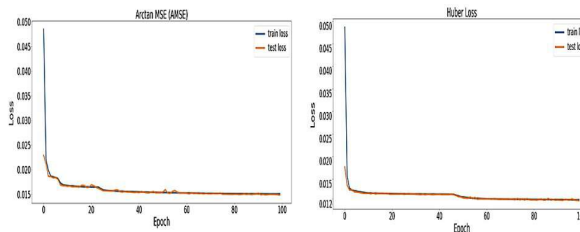


Figure 9: (a) Reconstruction error using Arctan MSE (b) Reconstruction error using Huber Loss. Reconstruction error comparison of proposed model

the AUC score obtained for the FANN model using HL 0.98 is compared to AMSE 0.96.

#### 4.4.2 Loss Functions: AMSE and HL

The proposed FANN model experiments with two loss functions: AMSE and HL during the training and test phase. The model’s training loss and test loss are accumulated for examining loss functions’ average reconstruction error. From the result of test loss, the threshold is varied to obtain outliers accurately and estimated with a range between 0.01 – 0.02.

Figure 4.4.2(a) and 4.4.2(b) display the reconstruction error obtained by the model after varying threshold for 100 epochs. From the above graph, it is evident that the FANN model with the HL function reacts more robustly to outliers for training and test data than the AMSE loss function. This resulted in a False Alarm Rate of 32% using AMSE and 16% using HL respectively.

#### 4.4.3 Conformance threshold

The conformance threshold value is calculated from table 2 which represents the mean and standard deviation of the model’s reconstruction error.

## 5 Results Comparison and Discussions

The proposed FANN model’s performance is evaluated in terms of success and competence. It has been compared with existing unsupervised algorithms as discussed in section 2. Figure 4.4.3 shows the

Table 2: Thresholding Comparisons using mean and standard deviation of the anomalies detected

Reconstruction error	AMSE	Huber Loss
Mean ( $\mu$ )	0.10211	0.09656
Standard Deviation ( $\sigma$ )	0.09061	0.08506
min	0.000021	0.000003
25%	0.000521	0.000116
50%	0.000822	0.000291
75%	0.001372	0.000699
max	0.296708	0.240890
# of False Positives	504	220

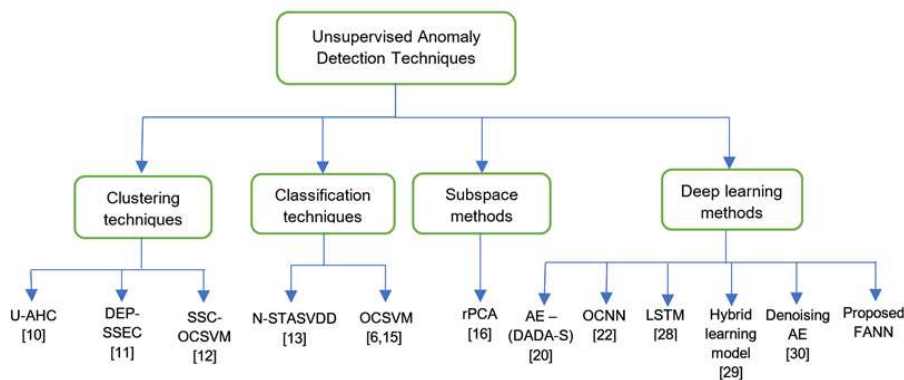


Figure 10: Categorization of unsupervised algorithms for comparative analysis

categorization of various algorithms such as clustering techniques U-AHC [10], DEP-SSEC [11], SSC-OCSVM [12], classification techniques N-STASVDD [13], OCSVM [6, 15], subspace method rPCA [16], and deep learning models AE-(DADA-S) [20], OCNN [22], LSTM [28], Hybrid learning model [29], Denoising AE [30].

Figure 5 shows the AUC score that is standardized for the  $10 \leq k \leq 100$  range. ‘k’ indicates the batch size of data instances. In our evaluation, many different ‘k’ values are considered and averaged by computing the AUC mentioned in table 3.

Environment: Experiments were run on a 4-core processor in Windows OS with Intel Core i7-8th Gen@1.8 GHz and 8GB RAM.

From Figure 5 it is observed that the proposed FANN achieves better reconstruction performance

Table 3: Analyses of AUC score of proposed FANN with unsupervised algorithms

Techniques	Method	AUC Score
Clustering	U-AHC	0.84
	SSC-OCSVM	0.88
	DEP-SSEC	0.77
Classification	N-STASVDD	0.92
	OCSVM	0.94
Subspace method	rPCA	0.95
Deep Learning	AE – (DADA-S)	0.92
	OCNN	0.89
	Proposed FANN Model (i) Arctan loss	0.96
	(ii) Huber loss	0.98

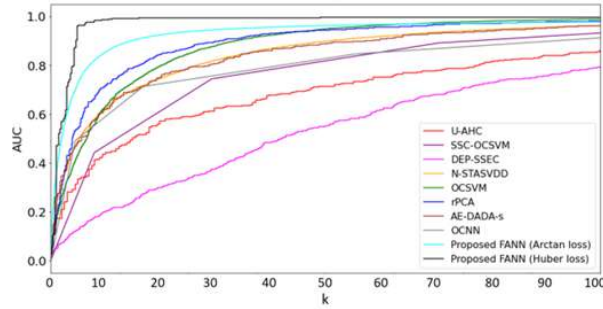


Figure 11: Generated AUC curves with different k values  $10 \leq k \leq 100$

with a higher AUC score for AD, which attempts to reduce RG (ii).

### 5.1 Anomaly Detection performance

The performance metrics selected for evaluation are Accuracy, Precision, Recall (or Sensitivity), Specificity, F1 Score, False Alarm Rate (FAR), and Matthews Correlation Coefficient (MCC) based on a confusion matrix representing the number of true positive rates (t.p rates), true negative rates (t.n rates), false-positive rates (f.p rates) and false-negative rates (f.n rates). The following equations are utilized for calculating the metrics,

$$\begin{aligned}
 Accuracy &= \frac{\#(t.prates+t.nrates)}{\#(t.prates+t.nrates+f.prates+f.nrates)} \\
 Precision &= \frac{\#(t.prates)}{\#(t.prates+f.prates)} \\
 Recall(or)Sensitivity &= \frac{\#(t.prates)}{\#(t.prates+f.nrates)} \\
 Specificity &= \frac{\#(t.nrates)}{\#(t.nrates+f.prates)} \\
 F1Score &= \frac{2*\#(t.prates)}{\#(2*t.prates+f.prates+f.nrates)} \\
 FalseAlarmRate(FAR) &= \frac{\#(f.prates)}{\#(f.prates+t.nrates)} \\
 Mathhewscorrelationcoefficient &= \frac{\#(t.prates*t.nrates) - \#(f.prates*f.nrates)}{\sqrt{(\#(t.prates+f.prates)*\#(t.prates+f.nrates)*\#(t.nrates+f.prates)*\#(t.nrates+f.nrates))}}
 \end{aligned}$$

For comparing the accuracy of our proposed model, experiments with different test sets are been carried out. The training data and test data are in the ratio of 3:2. To measure the accuracy of the models, in the 40% of test data various proportions of abnormality ranging from 1% to 15% are used. Table 4 shows the average accuracy calculated for each technique with various test window sizes. Based on experimentation, the accuracy of the proposed FANN model produces an improved accuracy by >98% making it suitable for AD in cluster head(CH) comparative to other existing techniques. Table 5 shows the evaluation results related to the precision, recall, specificity, F1 score, and False Alarm Rate (FAR) of the proposed FANN with other unsupervised algorithms discussed in the related work.

Figure 5.1 depicts the overall evaluation results of the models for AD. The results show that our proposed FANN yields higher detection performance. This addresses RG (i).

The motive of our proposed model is to design a false positive reducer model capable of reducing unnecessary false positives and avoiding data loss. This can be noticed in Figure 5.1(a), where the proposed FANN using HL produces lower false positives when compared with existing algorithms discussed in section 2, thereby addressing RG (iv). Finally, the metric Matthews Correlation Coefficient (MCC) indicates that the model produces good results only if it classifies both positive and negative elements. In Figure 5.1(b), the algorithms that produce lower MCC convey a random guess model whereas the proposed FANN has produced an MCC of 0.8 showing upright performance in unsupervised anomaly detection. This is a positive move towards reducing the RG (v). For an imbalanced

Table 4: Average accuracy evaluation for AD with different episodes (training set and test set)

Techniques	Method	Average Accuracy (%)
Clustering	U-AHC	0.84
	SSC-OCSVM	0.88
	DEP-SSEC	0.77
Classification	N-STASVDD	0.92
	OCSVM	0.94
Subspace method	rPCA	0.95
Deep Learning	LSTM	91.0
	Hybrid learning	92.4
	Denosing AutoEncoder	95.4
	AE – (DADA-S)	0.92
	OCNN	0.89
	Proposed FANN Model (i) Arctan loss	0.96
	Proposed FANN Model(ii) Huber loss	0.98

Table 5: Result of proposed models' efficiency with Sensitivity, Specificity, F1 Score, FAR and Mathews Correlation Coefficient (MCC)

Methods	Precision	Sensitivity	Specificity	F1 Score	FAR	MCC
U-AHC	92.6	90.8	11.5	91.7	88.5	0.02
SSC-OCSVM	90.7	92	9	91.3	90.7	0.01
DEP-SSEC	95.5	91.5	17	93.5	83.3	0.06
N-STASVDD	94.8	94.4	20	94.6	80	0.14
OCSVM	96.9	96.4	30	96.7	70	0.25
rPCA	98.7	98.7	40	98.7	60	0.38
LSTM	95.8	96.2	64	92.7	51	0.54
Hybrid learning	96.9	89.7	72	97.4	33	0.62
Denosing Autoencoder	97.5	95.4	80	96	39.2	0.49
AE – (DADA-S)	99	98.9	55.8	98.9	44	0.58
OCNN	98.1	98.5	30.7	98.3	69.2	0.32
FANN Model (i) Arctan loss	99.5	67.1	99.5	99.5	32.8	0.67
FANN Model (ii) Huber loss	99.7	83.6	99.8	99.8	16.3	0.8

dataset, analyzing the detection accuracy alone will not be a consistent measure. This exposes an over-optimistic estimation of the model in terms of efficiency. To get rid of the problem of uneven classification, the Matthews correlation coefficient (MCC) is estimated to identify the proposed model's capability concerning effectiveness. AD model must motivate on accurate detection of anomalies but all the compared algorithms such as,

- Clustering-based algorithms fail to enhance their accuracy due to their sensitivity to outliers,
- Classification and subspace methods lag in identifying true negatives and false positives for a large high-dimensional dataset with increased processing time,
- Deep learning algorithms fall back with lower MCC but produce higher false alarms, and require high computation time,

Therefore, the proposed FANN model can be identified as a robust model achieving expected detection performance for AD.

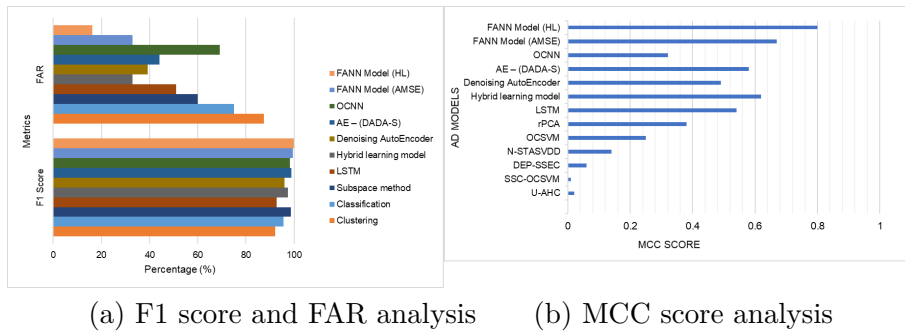


Figure 12: Comparison plot of significant performance metrics (F1 score, FAR, and MCC)

## 5.2 Energy Performance

In this section, to determine the utilization of energy during anomaly detection, complexity analysis is performed. Table 6 provides the complexity analysis used in similar approaches [40]. The computational complexity of FANN is  $O(rng)$  based on the evaluation of ‘r’ - number of inputs at the time, ‘n’ - total dimension of the instances, and ‘g’ - number of aggregators and the communication overhead obtained is  $O((r-a)n)$  where ‘a’ - number of abnormal instances that are removed as it uses deep learning model. The energy performance of sensor nodes for our work is evaluated based on the

Table 6: Complexity Analysis of Anomaly Detection

Techniques	Computational Complexity	Communication Complexity
Classification	$O(rn^2)$	$O(rvn^2)$
Clustering	$O(rn^6g)$	$O(rn^3)$
Subspace method	$O(r^3g)$	$O(r^3n)$
Deep Learning	$O(rng)$	$O((r-a)n)$

following factors [41]:

- Computing energy – the energy consumed by CH for data processing and analysis
- Communicating energy – the energy consumed by CH for receiving, and transmitting data with AD and without AD

A decentralized algorithm was discussed in [42] that computes energy in the dynamic network by developing a network coding subgraph that satisfies user-defined quality and security constraints. The Efficient Data Collection Aware of Spatio-Temporal Correlation (EAST) algorithm for energy-aware data forwarding in WSNs [43] takes full advantage of both spatial and temporal correlation mechanisms to save energy while maintaining real-time, accurate data reports toward the sink node. To minimize the overall energy consumption, a dynamic optimization model [44] developed multi-hop topologies to mitigate data transmission costs. For AD energy consumption, the following are the premises considered: Let  $E_{tot}$  be the initial energy unit of each aggregator node,  $E_{trn}$  be the energy unit of data packet communication,  $E_{rec}$  be the energy unit for receiving a data packet, and  $E_{ad}$  be the energy utilized by aggregator nodes ‘g’ for anomaly detection [45]. Then the residual energy  $E_{rd}$  of sensor node after reception can be expressed as,

$$E_{rd} = E_{tot} - E_{rec} \tag{7}$$

And the energy consumed by  $E_{ad}$  can be calculated as,

$$E_{ad} = C_{ON} \cdot T_{pc} \tag{8}$$

where,

$C_{ON}$  – Power consumed to start the processor

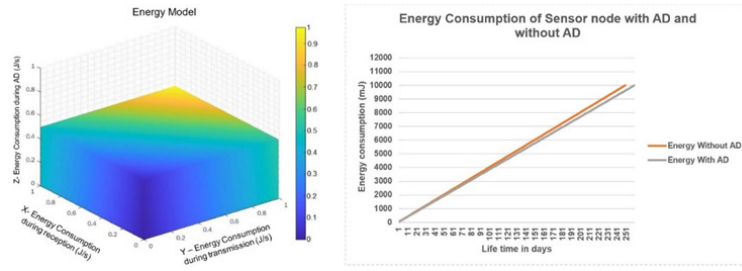


Figure 13: Energy Model for sensor nodes

$T_{pc}$ - The time utilized by the number of instructions and the microcontroller operating frequency,  $\frac{(\#ins)}{fq}$ .

Now, the remaining energy  $Et_{xrd}$  available for transmission is expressed as,

$$Et_{xrd} = E_{rd} - E_{ad} \tag{9}$$

Here the equation 9 states that there will be energy consumption due to  $E_{ad}$  before transmitting the aggregated data. To infer minimum energy consumption by  $E_{ad}$ , the following expectations are considered:

The total energy consumed by an aggregator node (sensor node)  $E_{tot}$  is measured with  $E_{rec} = (n * X)J/s$ ,  $E_{trn} = (m * Y)J/s$  and  $E_{ad} = (a)J/s$  as the assumed energy consumption unit. The value of ‘n’ indicates number of data instances sent to aggregator node and ‘m’ indicates number of data instances transmitted from aggregator node. The value of ‘m’ may vary or remains the same as ‘n’ depending on number of anomalies detected. A comparison of energy consumption of aggregator node is expressed in the following manner as:

- Considering the energy consumption without AD for aggregator node,

$$E1_{tot} = P_{st}T_{st} + E_{rec} + E_{trn} = P_{st}T_{st} + (n * X + m * Y)J/s \tag{10}$$

Here,  $E_{trn} = E_{rec}$  since the number data transmitted will be the same as received.

- Considering the energy consumption with AD for aggregator node,

$$E1_{tot} = P_{st}T_{st} + E_{rec} + E_{trn} + E_{ad} = P_{st}T_{st} + (n * X + (m - o) * Y + a)J/s \tag{11}$$

Here,

$E_{trn} = (m - o) * Y$  means energy consumed will be based on number data instances removed after AD.

$P_{st}T_{st}$  is the startup power and startup time for aggregator node

$o$  indicates the number anomalous data instances.

An energy model is built for equation (11) and simulated using MATLAB shown in Figure 5.2. This energy model predicted the lifetime of node energy with and without AD. It can be observed from the X axis that the lifetime of sensor nodes increases with AD. Illustration:

Let 0.004 J/s, 0.01 J/s and 0.01 J/s are the assumed energy consumption units for processing, data receiving and data transmission. Suppose the number of data instances received ‘n’ by aggregator node A1 is 100 then energy consumed by

$E_{rec}$  and  $E_{ad}$  will be,

$$E_{rec} = 100 * 0.01 = 1J/s$$

$$E_{ad} = 100 * 0.004 = 0.4J/s$$

Let the number of anomalies detected ‘o’ for aggregator node A1 is 10 then energy consumed by  $E_{trn}$  will be,

$$E_{trn} = (100 - 10) * 0.01 = 0.9J/s$$

Then  $E2_{tot}$  with 0.1 J/s as  $P_{st}T_{st}$  will be,

$$E2_{tot} = P_{st}T_{st} + E_{rec} + E_{trn} + E_{ad}$$

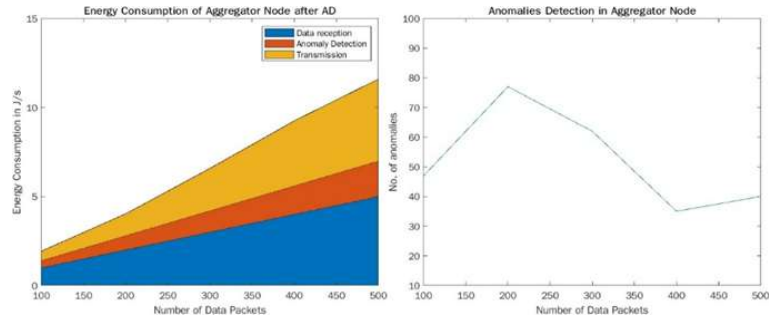


Figure 14: Energy consumption of aggregator node with AD using Energy Model

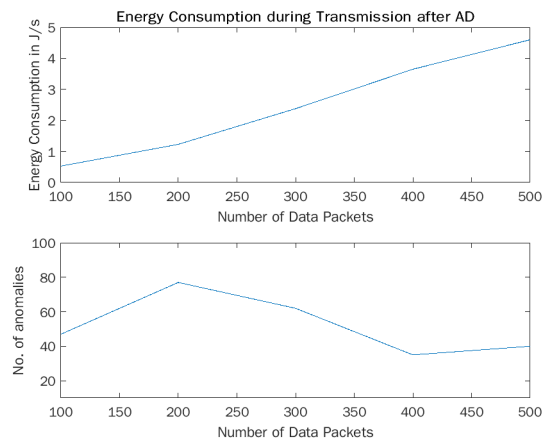


Figure 15: Energy consumption of transmission after AD using Energy Model

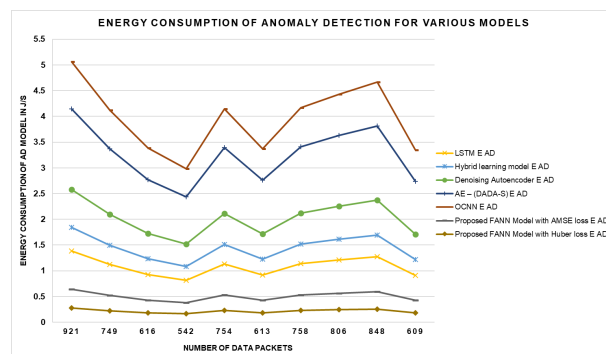


Figure 16: Comparison of average energy consumption of proposed AD model





Figure 17: Energy consumption analysis of various deep learning model

$$= 0.01 + 1 + 0.4 + 0.9J/s$$

$$= 2.31J/s$$

The above illustration expresses the reduction in transmission energy due to the removal of anomalous instances as depicted in Figure 5.2. Based on the premises discussed in the energy model shown in Figure 5.2, it can be inferred that FANN consumes proportionately less energy for AD than transmission and reception. The comparative analysis of energy consumed by AD for various models is depicted in Figure 5.2. From the comparison, it is identified that the proposed deep learning model consumes reduced energy than existing deep learning models and energy models for AD.

The computation of average energy consumption at a CH is analyzed. Following are the parameters used in our experiment:

- (i) Number of data packets is kept in a range of 150 to 750
- (ii) Data corruption rate is kept in a range of 1.5% to 15%

Also, the initial energy was set to 500 J/s and gradually decreased concerning processing various data packets using AD. A comparison of residual energy with AD against the residual energy without AD yields the inference. The energy consumption of various state-of-art techniques versus the proposed model is depicted in 5.2. The figure shows that the residual energy after AD using the proposed model appears to be marginally greater than the residual energy without AD. Other existing models, on the other hand, produce less residual energy after AD than residual energy without AD. This supports the proposed model for sustainable development in WSN. Consequently, the proposed FANN also ensures less computation time for AD compared to the other models as shown in 5.2, which is relatable to lower energy consumption, addressing RG (iii).

## 6 Conclusion

AD using deep learning is a significant research area that has attracted the attention of researchers. Specific techniques can be adopted depending upon the application domain. The detection process

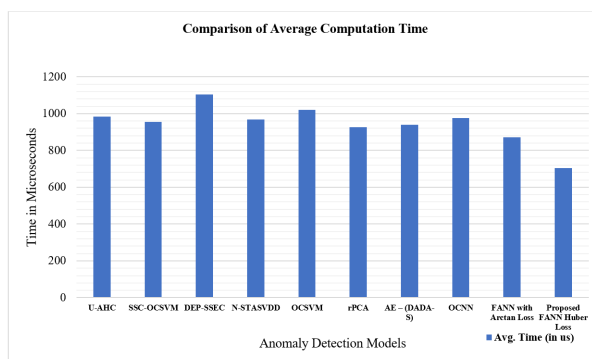


Figure 18: Computation time analysis of different unsupervised algorithms

must have high accuracy consuming less energy when it is applied to WSNs. The implemented FANN model presented in this paper is promising in AD handling unlabeled data gathered in a WSN environment. The selection of hyper parameters for the model provides a unique perspective for improving the accuracy of the model in eliminating outliers. Moreover, the proposed model is capable of handling high-volume multidimensional datasets with less computation time and reduced energy utilization. The performance of the proposed model is evaluated using the AUC-ROC score and suitable metrics for anomaly detection such as Accuracy, Precision, Recall, Specificity, F1 score, and MCC. From the experimental results, it is inferred that our model outperforms the other algorithms discussed in the literature concerning detection accuracy. Additionally, the proposed FANN model proves to be a False Positive Reducer achieving a lower FAR, which is a major concern for ensuring less data loss. The energy consumption of the proposed AD model is also observed as low in the aggregator node. The above discussions show that applying AD on the aggregator node will ensure sustainable improvement in WSNs. Furthermore, promising directions leading to extended research can focus on ensemble learning and auto-tuning the hyper parameters to further improve AD accuracy. Researchers can also work towards recovering lost data after an attack detection to empower data consistency in WSNs.

### Author contributions

The authors contributed equally to this work.

### Conflict of interest

The authors declare no conflict of interest.

### References

- [1] Miao Xie; Song Han; Biming Tian; Sazia Parvin (2011). Anomaly Detection in Wireless Sensor Networks: A survey, *Journal of Network and computer Applications*, vol. 34, no.4, pp. 1302-1325, 2011.
- [2] Deqing Wang; Ru Xu; Xiaoyi Hu; Wei Su (2016). Energy-Efficient Distributed Compressed Sensing Data Aggregation for Cluster-Based Underwater Acoustic Sensor Networks, *International Journal of Distributed Sensor Networks*, pp. 1-14, 2016.
- [3] Barakkath Nisha U; Uma Maheswari N; Venkatesh R; Yasir Abdullah R (2015). Improving Data Accuracy Using Proactive Correlated Fuzzy System in Wireless Sensor Networks, *KSII Transactions On Internet And Information Systems*, vol.9, pp.1976-7277, 2015.

- [4] Robert Mitchell; Ing-Ray Chen (2014). A survey of intrusion detection in wireless network applications, *Computer Communications*, vol. 2 no. 42, pp. 1-23, 2014.
- [5] Bo Sun; Xuemei Shan; Kui Wu; Yang Xiao (2013). Anomaly Detection Based Secure In-Network Aggregation for Wireless Sensor Networks, *IEEE Systems Journal*, vol. 7, no. 1, pp. 13-25, 2013.
- [6] Sapna Singh; Daya Shankar Singh; Shobhit Kumar (2014). Modified Mean Square Algorithm with reduced cost of training and Simulation time for Character Recognition in Backpropagation Neural Network, *Advances in Intelligent Systems and Computing*, Springer International Publishing, 2014.
- [7] XinqianLiu; Jiadong Ren; Haitao He; Qian Wang; Shengting Sun (2020). A Novel Network Anomaly Detection Method based on Data Balancing and Recursive Feature Addition, *KSII Transactions on Internet and Information Systems*, Vol. 14, No. 7, July 31, 2020.
- [8] Sankardas Roy; Mauro Conti; Sanjeev Setia; Sushil Jajodia (2012). Secure data aggregation in wireless sensor networks, *IEEE Information Forensics and Security*, vol. 7, no. 3, pp. 1040-1052, 2012.
- [9] Mohammed Abu Alsheikh; Shaowei Lin; Dusit Niyato; Hwee Pink Tan (2014). Machine learning in Wireless Sensor Network: Algorithm, Strategies & Application, *IEEE Communication surveys & tutorials*, vol. 16, 2014.
- [10] Francesco Gullo; Giovanni Ponti; Andrea Tagarelli; Sergio Greco (2017). An information-theoretic approach to hierarchical clustering of uncertain data, *Information Science*, 402, 199–215, 2017.
- [11] G. Yuan; B. Li; Y. Yao; S. Zhang (2017). A deep learning-enabled subspace spectral ensemble clustering approach for web anomaly detection, *International Joint Conference on Neural Networks (IJCNN)*, pp. 3896-3903, 2017.
- [12] Guo Pu; Wang L (2021). A hybrid unsupervised clustering-based anomaly detection method, *Tsinghua science, and technology*, ISSN 1007-0214 pp 146-153, vol. 26, No 2, 2021.
- [13] Chen. Y; Li. S (2019). A Lightweight Anomaly Detection Method Based on SVDD for Wireless Sensor Networks, *Wireless Personal Communication*, 105, 1235–1256, 2019.
- [14] Ruff. L; Vandermeulen. R; Goernitz. N; Deecke. L; Siddiqui. S.A; Binder. A; Müller. E; Kloft. M (2018) Deep One-class Classification, *Proceedings of the 35th International Conference on Machine Learning, Proceedings of Machine Learning Research*, 80:4393-4402,2018.
- [15] Nurfazrina Mohd Zamry; Anazida Zainal; Murad A. Rassam (2018). Unsupervised anomaly detection for unlabelled Wireless Sensor Networks Data, *International Journal Advance Soft Computing Applications*, Vol. 10, No. 2, 2018.
- [16] Xuehui Wang; Yong Zhang; Hao Liu; Yang Wang; Lichun Wang; Baocai Yin (2018). An Improved Robust Principal Component Analysis Model for Anomalies Detection of Subway Passenger Flow, *Journal of Advanced Transportation*, vol. 2018, 12 pages, 2018.
- [17] Aaron Tuor; Samuel Kaplan; Brian Hutchinson; Nicole Nichols; Sean Robinson (2017). Deep Learning for Unsupervised Insider Threat Detection in Structured Cybersecurity Data Streams, *Proceedings of AI for Cyber Security Workshop at AAAI*, 2017.
- [18] Marwan Ali Albahar; Muhammad Binsawad (2020). Deep Autoencoders and Feedforward Networks based on a New Regularization for Anomaly detection, *Security and Communication Networks*, Hindawi, 2020.
- [19] Raghavendra Chalapathy; Sanjay Chawla (2019). Deep Learning for Anomaly Detection – A Survey, *arXiv.org*, 2019.

- [20] T. Luo; S. G. Nagarajan (2018). Distributed Anomaly Detection Using Autoencoder Neural Networks in WSN for IoT, *IEEE International Conference on Communications (ICC)*, pp. 1-6, 2018.
- [21] Yan Qiao; Xinhong Cui (2020). Fast outlier detection for high-dimensional data of WSN, *International Journal of Distributed Sensor Networks*, vol. 16(10), 2020.
- [22] Raghavendra Chalapathy; Aditya Krishna Menon; Sanjay Chawla (2019). Anomaly Detection using ONE-CLASS NEURAL NETWORKS, *arXiv.org*, 2019.
- [23] Markus Goldstein; Seiichi Uchida (2016). A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data, *Journal.pone.0152173, PLOS ONE*, 2016.
- [24] DaochenZha; Kwei-Herng Lai; Mingyang Wan; Xia Hu (2020). Meta-AAD: Active Anomaly Detection with Deep Reinforcement Learning, *arxiv.org*, 2020.
- [25] F. de La Bourdonnaye; C. Teulière; T. Chateau; J. Triesch (2018). Learning of binocular fixations using anomaly detection with deep reinforcement learning, *International Joint Conference on Neural Networks (IJCNN)*, pp. 760-767, 2018.
- [26] Hoc Thai Nguyen; Nguyen Huu Thai (2019). Temporal & Spatial outlier detection in wireless sensor networks, *Wiley ETRI Journal*, 41(4):437–451, 2019.
- [27] Sahar Kamala; Rabie A. Ramadanb; Fawzy EL-Refai (2016). Smart Outlier Detection of WSN, *Facta universitatis - series: Electronics and Energetics*, vol. 29, Issue 3, pp. 383-393, 2016.
- [28] Jakovljevic; Mihajlo; Elbasani; Ermal; Kim; Jeong-Dong (2021). LLAD: Life-Log Anomaly Detection Based on Recurrent Neural Network LSTM, *Journal of Healthcare Engineering, Hindawi*, pp. 2040-2295, 2021.
- [29] S. Garg; K. Kaur; N. Kumar; G. Kaddoum; A. Y. Zomaya; R. Ranjan (2019). A Hybrid Deep Learning-Based Model for Anomaly Detection in Cloud Datacenter Networks, *IEEE Transactions on Network and Service Management*, 16, 3, 924-935, 2019, doi: 10.1109/TNSM.2019.2927886.
- [30] Chander. B; Kumaravelan (2021). Outlier Detection in Wireless Sensor Networks with Denoising Auto-Encoder, *Advances in Intelligent Systems and Computing*, 1382, 2021.
- [31] Chongxuan Li; Jun Zhu; Bo Zhang (2016). Learning to generate with memory, *In International Conference on Machine Learning (ICML)*, pp. 1177–1186, 2016.
- [32] Xavier Glorot; YoshuaBengio (2010). Understanding the difficulty of training deep feedforward neural networks, *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.
- [33] Szandała T (2021). Review and Comparison of Commonly Used Activation Functions for Deep Neural Networks, *In: Bio-inspired Neurocomputing. Studies in Computational Intelligence*, vol 903, 2021.
- [34] Nwankpa. C. E; Ijomah. W; Gachagan. A; Marshall. S (2021). Activation functions: comparison of trends in practice and research for deep learning, *2nd International Conference on Computational Sciences and Technology*, pp. 124–133, 2021.
- [35] Hendrycks. D; Gimpel. K (2016). Gaussian Error Linear Units (GELUs), *Machine Learning (cs.LG)*, *arXiv e-prints*, 2016.
- [36] Gokcesu; Kaan; Hakan Gokcesu (2021). Generalized Huber loss for robust learning and its efficient minimization for a robust statistics, *arXiv preprint arXiv:2108.12627*, 2021.
- [37] Vallez. N; Velasco-Mata. A; Deniz. O (2021). Deep autoencoder for false positive reduction in handgun detection, *Neural Computing & Applications*, 33, 5885–5895, 2021.

- [38] Bezerra. F; Wainer.J (2012). A Dynamic Threshold Algorithm for Anomaly Detection in Logs of Process Aware Systems, *Journal Information and Data Management*, vol. 3, p.316-331, 2012.
- [39] <https://www.wunderground.com/history/monthly/in/new-delhi/VIDP>
- [40] Barakkath Nisha Usman; Uma Maheswari Natarajan; Venkatesh Ramalingam; Yasir Abdullah Rabi (2016). Fuzzy based Flat Anomaly Diagnosis and Relief Measures in Distributed Wireless Sensor Network, *International Journal of Fuzzy Systems*, vol 19, 2016.
- [41] John. T; Ogbiti; Ukwuoma Henry; Danjuma Salome; Ibrahim Mohammed (2016). Energy Consumption in Wireless Sensor Network, *The International Institute for Science, Technology, and Education (IISTE)*, 2016.
- [42] Mohajer. A; Mazoochi. M; Niasar. F.A; Ghadikolayi. A.A; Nabipour. M (2013). Network Coding-Based QoS and Security for Dynamic Interference-Limited Networks, *Communications in Computer and Information Science*, 370, 2013.
- [43] Leandro A. Villas; Azzedine Boukerche; Daniel L. Guidoni; Horacio A.B.F. de Oliveira; Regina Borges de Araujo; Antonio A.F. Loureiro (2013). An energy-aware Spatio-temporal correlation mechanism to perform efficient data collection in wireless sensor networks, *Computer Communications*, 36, 9, 2013.
- [44] A. Mohajer; F. Sorouri; A. Mirzaei; A. Ziaeddini; K. J. Rad, M. Bavaghar (2022). Energy-Aware Hierarchical Resource Management and Backhaul Traffic Optimization in Heterogeneous Cellular Networks, *IEEE Systems Journal*, pp. 1-12, 2022. doi: 10.1109/JSYST.2022.3154162.
- [45] Mohamed Elshrkawey; Samiha M. Elsherif; M. ElsayedWahed (2018). An Enhancement Approach for Reducing the Energy Consumption in Wireless Sensor Networks, *Journal of King Saud University – Computer and Information Sciences*, vol. 30, 259–267, 2018.



Copyright ©2023 by the authors. Licensee Agora University, Oradea, Romania.

This is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial 4.0 International License.

Journal's webpage: <http://univagora.ro/jour/index.php/ijccc/>



This journal is a member of, and subscribes to the principles of,  
the Committee on Publication Ethics (COPE).

<https://publicationethics.org/members/international-journal-computers-communications-and-control>

*Cite this paper as:*

Arul Jothi, S.; Venkatesan, R. (2023). A Deep Learning Approach for Efficient Anomaly Detection in WSNs, *International Journal of Computers Communications & Control*, 18(1), 4756, 2023.

<https://doi.org/10.15837/ijccc.2023.1.4756>