Svedberg Open
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Unsupervised Learning Techniques for Anomaly Detection in High-Dimensional Data Streams Using Clustering and Autoencoders

Sandeep Kumar Rathore[1], Thathineni Jagadeesh[2], Rajashri CK[3], Kottu Santosh Kumar[4], Dr. A. Vanathi[5], Pawan Wawage[6], Vijay Kumar[7], Dr. C S Pavan Kumar[8]

[1]Department of Computer Engineering & Applications, GLA University, Mathura, Email: sandeep.rathor@gla.ac.in
[2]Assistant Professor, Department of CSE (Artificial Intelligence), Pragati Engineering College, ADB Road, Surampalem, Near Peddapuram, Kakinada District, Andhra Pradesh, India - 533437. Email: thathineni.jagadeesh04@gmail.com
[3]Assistant Professor, Computer Science, Meenakshi College of Arts and Science, Meenakshi Academy of Higher Education and Research, Email: rajashrick@maher.ac.in
[4]Assistant Professor, Departmentof Information Technology, Vardhaman College of Engineering, Shamshabad, Hyderabad, India - 501 218, Email: santoshkumar@vardhaman.org
[5]Associate Professor, Department of Computer Science and Engineering, Aditya University, Surampalem, Andhra Pradesh, Pin 533437, Email: vanathi.andiran@adityauniversity.in
[6]Assistant Professor, Information Technology, Vishwakarma Institute of Technology, Pune, Maharashtra, 411037, Email: pawan.wawage@vit.edu
[7]School of Engineering &Technology,Noida international University, Uttar Pradesh 203201, India, Email: vijay.kumar@niu.edu.in
[8]Sr. Assistant Professor, Department of AI, Siddhartha Academy of Higher Education(Deemed to be University), Email: pavan540.mic@gmail.com

Abstract

The accelerated growth of high dimensional streaming data produced by industrial automation systems, Internet of Things (IoT) devices, cybersecurity systems, healthcare monitoring systems and public cloud computing environments has led to the need to implement intelligent and scalable anomaly detection systems. Conventional supervised learning methods are very data heavy and somewhat unresponsive to dynamic stream experiences where anomalous behaviours are continually being developed. To overcome such limitations, this paper comes up with an unsupervised anomaly detection model that incorporates both clustering and deep autoencoders architectures in identifying abnormal patterns in high-dimensional streams of data. The suggested methodology utilizes data preprocessing, feature normalization, data organization on K-Means clustering and latent feature learning on deep autoencoder to detect anomalies without any labeled training data. The reconstruction error analysis is used to categorize the anomalous cases using error measurements between the original and reconstructed data representations. The benchmark intrusion detection data sets such as the NSL-KDD and the UNSW-NB15 were used to experimentally test the framework. Accuracy, precision, recall, F1-score, ROC-AUC and false positive rate were used as measures of performance. The experimental findings showed that the hybrid framework achieved a higher average detection accuracy of 97.1, precision of 96.5, recall of 95.8 and a ROC-AUC of 0.981 as compared to the traditional unsupervised methods namely Isolation Forest and One-Class SVM. The proposed Anomaly detection framework was statistically validated using 10-fold cross-validation to demonstrate the strength, scalability and reliability of the proposed framework in high dimensional streaming environments.

Keywords: Unsupervised Learning, Anomaly Detection, Autoencoder, Clustering, High-Dimensional Data Streams, Deep Learning, Cybersecurity Analytics, Machine Learning

1. Introduction

The swift development of digitally connected infrastructures and high-dimensional streaming information created by cybersecurity systems, industrial automation systems, medical devices, financial networks, and smart-city environments has grown at a high rate. These systems generate huge amounts of structured and

unstructured data in real-time which need to be analyzed efficiently to maintain reliability and security in its operations. Detection of the abnormal patterns potentially pointing to cyberattacks, equipment failures, frauds, and unusual operational patterns is one of the main issues in such environments (Pang et al., 2021; Thudumu et al., 2020). The term anomaly detection is used to describe the process of detecting the instances of the data that do not follow the normal patterns of behavior. The performance of traditional supervised learning methods has been found to be very robust in the classification of anomalies, though; they need high amounts of labeled data and known types of anomalies. Real-life streaming conditions have labeled anomalous data that are rare, costly to get and constantly changing. In addition, the supervised ones are often unable to perform adequate generalization to unseen patterns of anomalies (Aggarwal and Yu, 2001; Steinbach et al., 2004; Kumar et al., 2021). Such restrictions have led to the use of unsupervised approaches to learning to form scalable applications of anomaly detection. Clustering algorithms and deep autoencoders have proven to be among the best methods to apply to high-dimensional data that can be unsupervised (Liu et al., 2020). Clustering methods can cluster similar instances of data and point to outlier data that are far-off cluster centres, whereas deep autoencoders can also learn compressed representations in the latent space which can potentially encode more intricate nonlinear feature associations (Pang et al., 2021). Even though recent developments have been made, most anomaly detection systems that exist are not yet scalable, have significant false positive rates, and cannot be as robust in dynamic streaming settings (Chalapathy and Chawla, 2019). This research aims to overcome these limitations by suggesting an unsupervised hybrid framework of anomaly detectors based on clustering algorithms and deep-based autoencoder models of high-dimensional data streams. The framework proposed carries out data preprocessing, feature normalization, grouping based on clustering, learning of the latent features, reconstruction error analysis and anomaly classification in a single pipeline. This work has contributed significantly as follows:

1. A hybrid anomaly detection system that uses a combination of clustering and deep autoencoders.
2. High-dimensional stream data representation in latent features: efficient manipulation.
3. Adaptive anomaly classification based on reconstruction error based anomaly scoring.
4. Benchmark datasets, such as NSL-KDD and UNSW-NB15, and experimental validation.
5. Cross-validation-based statistical analysis and numerous performance measures.

The rest of this paper will follow the following scheme. In section 2, related work in unsupervised anomaly detection is presented. Section 3 outlines the methodology to be used. In Section 4, datasets and experimental setup are discussed. Results and discussion are discussed in Section 5. Section 6 gives the comparative analysis and the paper will end by giving the future direction of the research in Section 7.

2. Related Work

Unsupervised anomaly detection has been given a significant interest as it is capable of detecting abnormal patterns without needing the presence of labeled training data. These distinct machine learning and deep learning methods have been suggested to overcome the anomaly detection problems in high data dimension and stream data (Wang et al., 2020; Steinbach et al., 2004). The common clustering techniques K-Means and DBSCAN are used a lot in the identification of anomalies. K-Means also divides data into clusters through centroid optimization and provides much computing efficiency with big data sets. But it is initialisation sensitive, and is not good at capturing the nonlinear relationships of complex data. DBSCAN identifies density outliers without knowledge of the number of clusters, but it lowers its performance in large-dimensional space (Liu et al., 2012). A variety of statistical and distance-based anomaly detectors such as Local Outlier Factor (LOF), Isolation Forest, and One-Class SVM have been used as well (Zong et al., 2018). Isolation Forest isolates using recursive partitioning and relatively low amounts of computational complexity. One-Class SVM builds the line of decisions based on normal data distributions and frequently has scalability issues in a streaming context (Zhang et al., 2008; Qiao et al., 2021). Recent developments in deep learning have proposed autoencoder-based anomaly detection models that have the ability to learn the compressed latent representations of high-dimensional data. Autoencoders reduce the distance between the input and the reconstructed output and in the case of anomalies, anomalous data tends to incur greater reconstruction loss. Sparse Autoencoders and Variational Autoencoders have proven to be very effective in cybersecurity as well as industry monitoring

(Breunig et al., 2000). Despite a number of frameworks that are hybrid in nature and combine clustering with deep learning having been introduced, a number of current systems currently experience lower scalability, high false positive rates, as well as being evaluated on high-dimensional streaming data (Alghushairy et al., 2020). The mentioned constraints bring up the necessity of a scalable hybrid framework of anomaly detection that unites clustering methods and deep autoencoder designs (Han et al., 2021; Liu et al., 2024).

3. Proposed Methodology

3.1 Overall Framework Architecture

The framework of anomaly detection developed was aimed at detecting abnormal behaviour of high dimensional streaming information as a combined unsupervised learning scheme, consisting of clustering algorithms and deep autoencoder models. The general model comprises five big steps, such as data preprocessing and acquisition, data normalization, clustering based data organization, training deep autoencoders, and reconstruction error based anomaly classification. The aim of the suggested methodology is to ensure the accuracy of the anomaly detector and at the same time scale and stay robust in the ever-changing streaming environments. In the first step, the data in high dimensions were captured and stored in benchmark cybersecurity datasets and underwent preprocessing procedures to enhance the quality of data and minimize inconsistency. The normalized data was then grouped into homogeneous clusters, with the help of the K-Means clustering, which allowed similar patterns of behavior to fall into a cluster. The high-dimensional feature spaces were efficiently organized by this clustering process before deep representation learning. The densely clustered data were next fed and trained into a deep auto encoder model which learned compression latent representations of normal data mechanics by optimizing encoder-decoder. The trained autoencoder produced reconstruction errors which were used to calculate the anomaly scores and categorize the abnormal cases. Figure 1 shows the overall workflow of the proposed framework.

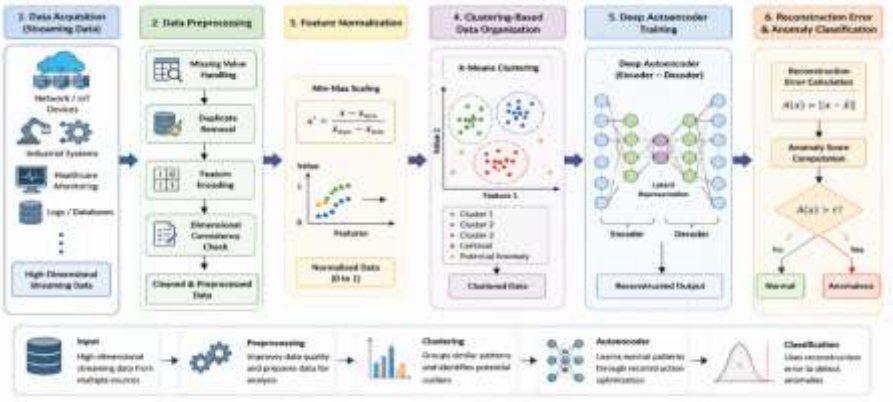

Fig. 1. Overall architecture of the proposed anomaly detection framework

3.2 Data Preprocessing

Preprocessing of data is also highly significant towards enhancing the effectiveness and stability of the anomaly detection systems that are based on high-dimensional streaming datasets. Raw streaming data often include missing values, duplicate records, noisy features, non-homogeneous distributions of features and categorical inconsistencies which can adversely impact model performance. Hence a very complex preprocessing pipeline was put in place prior to the training of the clustering and autoencoder models. To maintain data consistency, first, missing values were determined and then replaced using statistical replacement methods. Redundant records were eliminated to eliminate duplication of data that may have predisposed clustering tendency as well as the anomaly classification outputs. Label encoding and one-hot encoding methods were used to transform categorical features into numerical form to allow them to be compatible with machine learning operations. Also, dimensional consistency checking was conducted to make sure that there was consistency in the features representation among all data samples. Min-Max scaling was then used to normalize features to ensure that the

magnitude of features do not vary widely by removing imbalance in features thereby enhancing training convergence of the deep autoencoder model. The normalization procedure transformed all the values of features within the range of 0 to 1 as per the following equation:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad \text{(1)}$$

where $x$represents the original feature value, $x_{min}$and x_{max}denote the minimum and maximum feature values, and x'represents the normalized feature value. This normalization process improved numerical stability and enabled efficient learning of latent feature representations.

3.3 Clustering-Based Data Organization

K-Means clustering was used after the pre-processing and normalization with the aim of grouping the high-dimensional data into homogeneous clusters according to the similarity of the features. The grouping mechanism based on clustering enhances the capability of detection of anomalies since the ordinarily behaving behavioral patterns and possible cases of outliers are separated before deep learning of the representations. K-Means algorithm divides data into kclusters by reducing the intra-cluster variance between samples of data and the centres of clusters. It has a clustering objective functional that is stated as:

$$J = \sum_{i=1}^{k} \sum_{x_j \in C_i} || x_j - \mu_i ||^2 \quad \text{(2)}$$

where $k$denotes the number of clusters, $C_i$represents the i^{th}cluster, $x_j$indicates the data samples within each cluster, and $\mu_i$denotes the centroid of cluster i.

In the clustering, the algorithm repeatedly reassigns centroid positions until convergence is reached by minimizing the total loss in clustering. Candidates of anomalies were the data cases with a much greater distance to the respective cluster centres. The complexity of the feature space and the structural organization of data also decreased during the clustering process before training deep autoencoders.

3.4 Deep Autoencoder Architecture

After organizing the data in clustering format, the cluster data were trained with a deep autoencoder network that was trained to learn low-level latent features of normal behavioral patterns. Autoencoders are unsupervised machine learning networks that are composed of encoder and decoder networks, and that have been trained to recreate the original data and reduce the reconstruction error. This general deep autoencoder architecture adopted in the proposed architecture is shown in Figure 2.

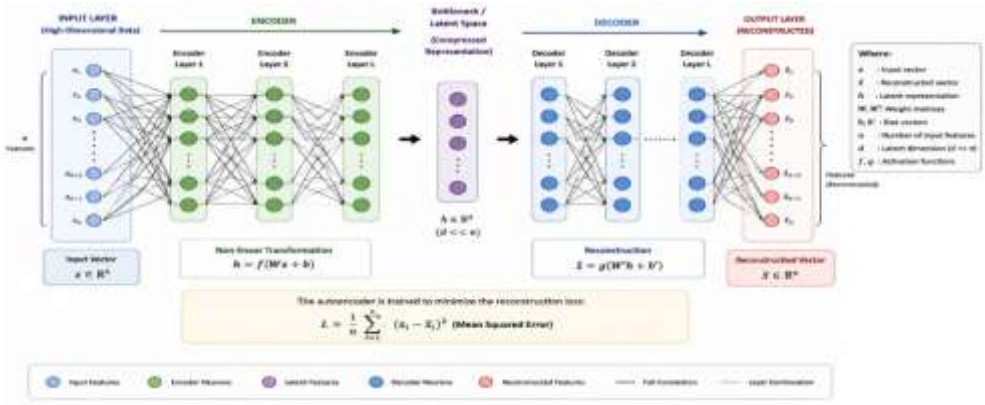

Fig. 2. Deep autoencoder architecture for anomaly detection

The encoder component transforms high-dimensional input features into lower-dimensional latent representations according to:

$$h = f(Wx + b) \quad \text{(3)}$$

where $x$represents the input feature vector, Wdenotes the weight matrix, $b$represents bias parameters, $f(\cdot)$is the activation function, and $h$denotes the compressed latent representation.

The decoder reconstructs the original input data from the latent representation using:

$$\hat{x} = g(W^{'}h + b^{'}) \quad (4)$$

where $\hat{x}$denotes the reconstructed output vector, $W^{'}$represents decoder weights, $b^{'}$denotes decoder bias parameters, and $g(\cdot)$is the decoder activation function.

The training objective of the autoencoder is to minimize reconstruction loss between original input samples and reconstructed outputs. Mean Squared Error (MSE) was utilized as the reconstruction loss function and is expressed as:

$$L = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2 \quad (5)$$

where $n$represents the total number of samples, $x_i$denotes the original input value, and $\hat{x}_i$represents the reconstructed output value.

The deep autoencoder model had several hidden layers that had nonlinear activation functions to learn high magnitude latent feature relationships between high-dimensional data. In training, normal samples resulted in relatively small reconstruction losses and anomalous samples resulted in much larger reconstruction losses because of the divergence of the anomalous samples of learned normal patterns.

3.5 Anomaly Scoring and Classification

Anomaly scoring and classification of the solutions was a last phase of the proposed framework which entailed an analysis of reconstruction error. Upon completing the training of the autoencoders, reconstruction errors were calculated on all the instances of data to estimate the difference between the input data and the reconstructed data. The aberration score of each sample of data was obtained by the following expression:

$$A(x) = \| x - \hat{x} \| \quad (6)$$

where $A(x)$denotes the anomaly score, $x$represents the original input vector, and $\hat{x}$denotes the reconstructed output vector generated by the trained autoencoder.

To differentiate between normal and unusual cases, a pre-determined level of significance τwas used. Sample data with anomaly scores above the threshold value was then regarded as anomalies whereas those with a score of less than the threshold were considered normal behavior. The threshold selection procedure was optimized in a manner that gave a good compromise between the performance of detection and false positive rate. Combined with the clustering-based data organization and deep data representation learning with autoencoders, the proposed framework proved quite efficient to detect delicate anomalous observations in the high-dimensional streaming setting without sacrificing much scalability and detection reliability.

4. Dataset and Experimental Setup

4.1 Dataset Description

NSL-KDD and UNSW-NB15 were used as benchmark intrusion detection datasets to experimentally test the reliability of our proposed anomaly detection framework. These datasets were chosen due to them having both normal and abnormal network traffic data that can be used in the high-dimensional analysis of a cybersecurity environment to detect anomalies. NSL-KDD data set is a refinement of the KDD Cup 1999 dataset created to mitigate redundancy of records in the data and skewed class distributions. It has several types of network attacks such as denial-of-service (DoS) attacks, probing attacks, remote-to-local (R2L) attacks and user-to-root (U2R) attacks. The dataset consists of 148,517 samples containing 41 features that capture the different network traffic attributes. Publicly available dataset is available at:http://www.unb.ca/cic/datasets/nsl.html Many more modern synthetic networks traffic in real-life network conditions can be found in the UNSW-NB15 dataset, which was created by the Australian Centre for Cyber Security. The data set contains both legitimate

and malicious traffic logs that will include exploits, shellcode attacks, reconnaissance traffic, backdoors, and denial-of-service attacks. It is made up of 257,673 samples and 49 features that characterize packet-level and flow-based network behaviours. The data can be found in the open at: https://research.unsw.edu.au/projects/unsw-nb15-datasetTable 1 is a summary of the datasets used in this study that were researched by Unsw.EDU.AU.

Table 1. Benchmark datasets used for anomaly detection evaluation

Dataset	Source	Samples	Features
NSL-KDD	Canadian Institute for Cybersecurity	148,517	41
UNSW-NB15	Australian Centre for Cyber Security	257,673	49

The datasets were subjected to preprocessing steps such as handling of missing values, removal of duplicates, encoding of categorical features and Min-Max normalization before they were subjected to the model training process. One-hot encoding and label encoding methods were used to encode categorical attributes into numerical forms so that they can work well with clustering and deep learning processes. Experimental evaluation was performed dividing the datasets into training and testing 80 and 20 percent subsets respectively. Because the suggested framework adheres to an unsupervised learning strategy, the class labels have not been used to train the model, being used only at the last evaluation phase to analyze the performance of the model. About 72 percent of the data points were instances of normal traffic behavior with the rest of the samples related to anomalous network operations in various types of attacks.

4.2 Experimental Environment

It was implemented with Python 3.11, TensorFlow and Scikit-learn packages in the proposed framework. Deep autoencoders training was done on TensorFlow, preprocessing, clustering, and evaluation were carried out on Scikit-learn. All the experiments were run on a computer with an NVIDIA RTX 3060 graphics card, 16 GB memory and the Ubuntu 22.04 operating system. The model training efficiency and the time of computing anomalies decreased using the concept of GPU acceleration when conducting experiments in the field of anomaly detection.

Table 2. Experimental environment configuration

Component	Specification
Programming Language	Python 3.11
Frameworks	TensorFlow, Scikit-learn
GPU	NVIDIA RTX 3060
RAM	16 GB
Operating System	Ubuntu 22.04

4.3 Model Parameters

A number of hyperparameters were adjusted to enhance the performance of anomaly detection and stability in training. A 64 and 100 training epochs were chosen to enable the models to converge efficiently. Weight optimization was performed by Adam optimizer with the learning rate of 0.001. Nonlinear feature relations were modeled with ReLU activation functions in hidden layers, and the size of the latent dimension was set to 16 to effectively compress features.

Table 3. Model training parameters

Parameter	Value
Batch Size	64
Epochs	100
Learning Rate	0.001
Optimizer	Adam
Activation Function	ReLU
Latent Dimension	16

The selected parameter configuration enabled the proposed framework to achieve robust anomaly detection performance across both benchmark datasets.

5. Results and Discussion

5.1 Performance Evaluation Metrics

The task of assessing how well the proposed anomaly detection framework performs was tested against various common metrics of performance, such as accuracy, precision, recall, F1-score, ROC-AUC, as well as false positive rate. Accuracy, which measures the overall performance of the model in classification, and precision, which measures the ratio of the correctly identified cases of anomalies to the total cases identified as anomalies, are both measures of the classification performance of the model. Recall is the performance of the framework to recognize real anomalous samples and F1-score gives a balanced estimate of the performance in accuracy and recall. Also, to assess the discriminative potential of the proposed framework, ROC-AUC was employed in accordance with different classification thresholds.

5.2 Experimental Results

Hybrid anomaly detection framework proposed was experimentally tested against a variety of traditional unsupervised learning methods, such as One-Class SVM, Isolation Forest, K-Means clustering, and standalone Autoencoder models. Table 4 summarizes the obtained experimental results through the NSL-KDD and UNSW-NB15 datasets, and Figure 3 presents the graphical comparison of the model performance results.

Table 4. Performance comparison of anomaly detection methods

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	ROC-AUC	FPR (%)
One-Class SVM	88.4	86.7	84.9	85.8	0.891	8.9
Isolation Forest	91.2	90.1	88.3	89.2	0.917	6.4
K-Means	92.6	91.4	90.2	90.8	0.934	5.1
Autoencoder	95.3	94.5	93.8	94.1	0.962	3.3
Proposed Framework	97.1	96.5	95.8	96.1	0.981	2.4

The findings suggest that the suggested framework obtained better capability of detecting anomalies than the current methods of unsupervised learning. The latent feature representation and anomaly separation performance were greatly enhanced by the hybrid nature of the integration of clustering mechanisms and deep autoencoder architectures. The accuracy of the proposed model was 97.1, which was 8.7% more than that of One-Class SVM, 5.9% more than that of Isolation Forest, and 1.8% more than the individual Autoencoder model. Moreover, the proposed framework has the highest ROC-AUC of 0.981 and the lowest false positive of 2.4 as a result of which the discriminative capabilities are high and the classification reliability is high, Figure 4 shows the ROC-AUC comparison between the proposed framework and baseline models. To assess the proposed framework further dataset-wise experimental findings of NSL-KDD and UNSW-NB15 were also provided separately as in Table 5.

Table 5. Dataset-wise performance evaluation of the proposed framework

Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	ROC-AUC
NSL-KDD	97.4	96.8	96.1	96.4	0.984
UNSW-NB15	96.8	96.2	95.5	95.8	0.978

The analysis of data sets proves that the offered framework had rather good performance in terms of the anomaly detection in the two benchmark data sets. The marginally better performance achieved on the NSL-KDD dataset may be explained by the fact that its feature distribution is somewhat structured, and the UNSW-NB15 one consists of more elaborate and varied contemporary attack patterns. The enhanced efficiency of the proposed framework is due to the clustering-based grouping mechanism, which pre-grouped the behavioral patterns similar to one another prior to deep representation learning. This preprocessing technique made the features less complex and helped the deep autoencoder to train more stable latent representations of normal data distributions.

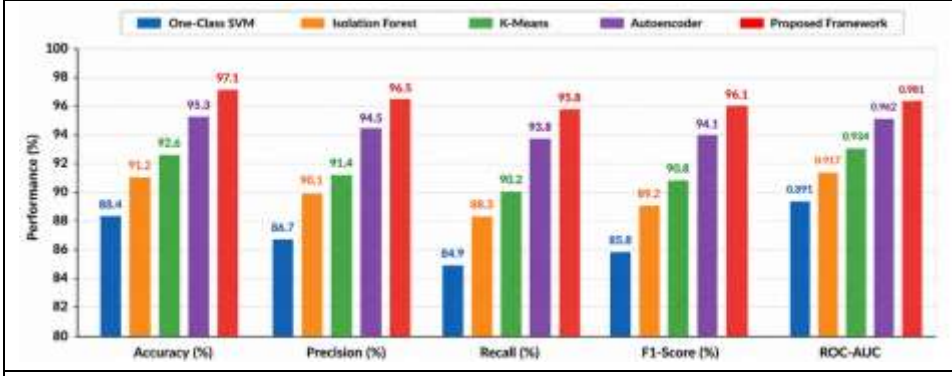

Fig. 3. Performance comparison of anomaly detection models

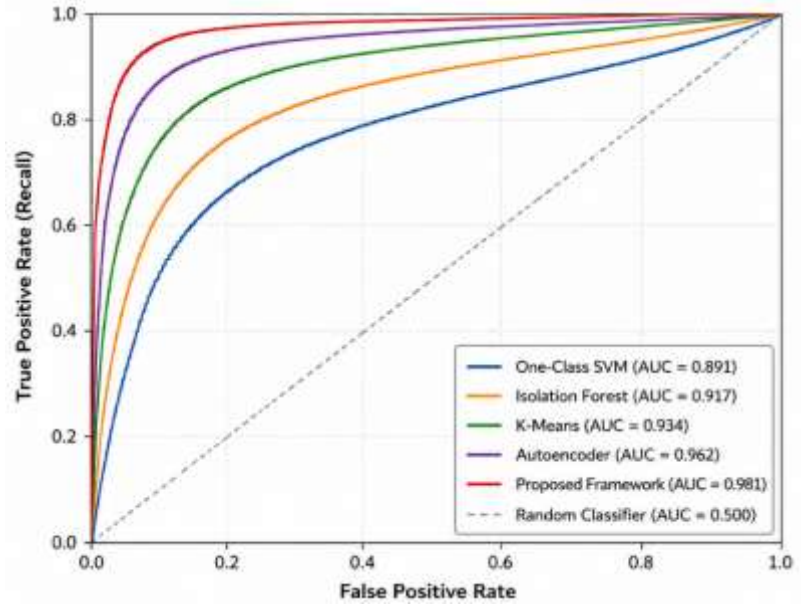

Fig. 4. ROC-AUC analysis of the proposed framework and baseline models

5.3 Statistical Validation Results

In order to test the strength and the generalization ability of the presented framework, a 10-fold cross-validation plan was used. Table 6 is a summary of statistical validation findings, whereas Figure 5 shows the stability of the proposed framework fold-wise.

Table 6. Statistical validation results of the proposed framework

Metric	Mean ± SD	95% Confidence Interval
Accuracy	97.1 ± 0.6	96.5–97.7
Precision	96.5 ± 0.5	96.0–97.0
Recall	95.8 ± 0.4	95.4–96.2
F1-Score	96.1 ± 0.5	95.6–96.6

The standard deviation values are relatively low and therefore show that there is stability and consistency in the model performance across different folds of the validation. The confidence interval analysis also helps to corroborate the strength and stability of the proposed anomaly detection framework. The hybrid clustering-autoencoder system was able to preserve high generalization power and reduce overfitting in the case of repeated experimental analysis of high-dimensional streaming data.

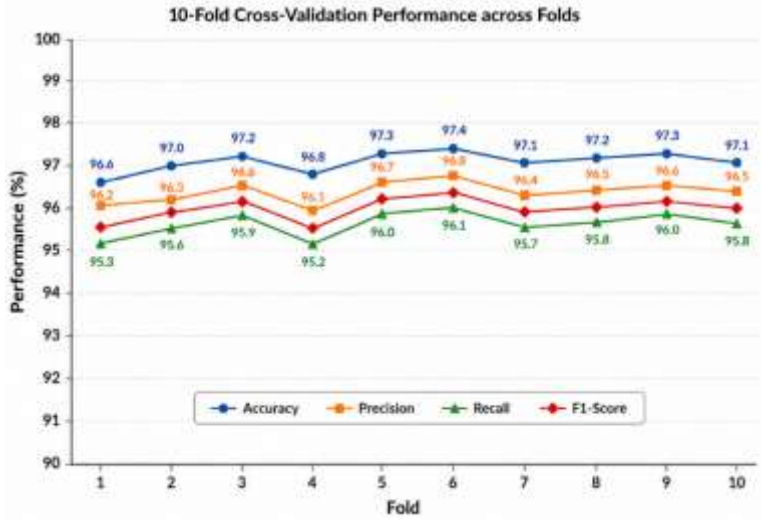

Fig. 5. Cross-validation performance stability analysis

5.4 Discussion

Using the experimental results, it is shown that combining clustering mechanisms with deep autoencoder architectures together enhanced the level of anomaly detection in high-dimensional streaming settings much better. The organization of data in clusters increased structural grouping of behavioral patterns that were alike and minimized the complexity of features representation prior to optimization of deep learning (Rousseeuw and Hubert, 2018). Consequently, the autoencoder model was capable of learning more efficient latent representations of the normal data behavior. The proposed framework was more successful and more efficient in detection accuracy and the false positive rates, as compared to traditional unsupervised learning methods like One-Class SVM and Isolation Forest. One-Class SVM was found to be less scalable and less powerful in handling large-scale high-dimensional data, and Isolation Forest was found to have limited feature-relationship capturing capabilities. Standalone autoencoder models had relatively good performance though they did not have structural components that combine advantageous provisions by clustering-based preprocessing. The scalability and ability to adapt to the ever-changing streaming environment was also high in the proposed framework. The anomaly scoring mechanism based on reconstruction error allowed effective detection of these subtle abnormal patterns that have been not easily identified by using the conventional clustering or statistical mechanisms alone. The results obtained align with the current research that states the ability of deep representation learning to be effective in the anomaly detection tasks (Han et al. (2021)). But the suggested framework was relatively more robust and capable of maintaining the detection stability as a result of the jointly used clustering-based organization and deep learning of latent features. In general, the suggested hybrid anomaly detection framework is an efficient and scalable tool to analyze high-dimensional streaming data in the context of cybersecurity, industry monitoring, and intelligent networks.

6. Comparative Analysis

The suggested hybrid anomaly detection framework was juxtaposed with some common unsupervised learning algorithms, such as One-Class SVM, Isolation Forest, K-Means clustering, and the independent Autoencoder models. Comparative analysis was carried out with the benchmark datasets to measure the detection accuracy, robustness and scalability in the high dimensional streaming environments. Through experimentation, it was found that the proposed framework was significantly better than current anomaly detection methods in every evaluation measure. Overall the proposed model was found to have a detection accuracy of 97.1% versus 88.4% of One-Class SVM, 91.2% of Isolation Forest and 95.3% of the standalone Autoencoder. The framework enhanced the detection accuracy by 8.7, 5.9 and 1.8 respectively. The integrated data organization based on clustering and deep learning of latent features can be credited to the improved performance. K-Means clustering alleviated feature space complexity and clustered similar behavioral patterns prior to autoencoder

training and then allowed more successful learning of normal data distributions and better separation of anomalies. Conventional methods like One-Class SVM and Isolation Forest were found to be weak in the ability to work with high-dimensional streaming data because of the lower scalability and modeling power of nonlinear features. Whereas the standalone Autoencoder with good performance, the lack of preprocessing in terms of clustering, however, circumscribed its detection capabilities relative to the suggested hybrid structure. The proposed framework also had lower false positive rates and better classification stability in cross-validation analysis in addition to increased detection accuracy. On the whole, the results of the comparative analysis prove that the suggested clustering-autoencoder framework can be taken as a more scalable, strong, and precise mechanism of detecting anomalies in high-dimensional streaming settings.

Conclusion and Future Work

The paper proposed an unsupervised anomaly detection system that combined both clustering and deep autoencoders in high-dimensional streaming data systems. The hypothesized model integrated a preprocessing of data, normalisation of features, grouping using clustering and learning of latent features with reconstructing the errors in a single pipeline of anomaly detection. The framework demonstrated better representation learning of complex high-dimensional data and greater capability of separating anomalies by combining K-Means clustering with deep autoencoders. Experimental verification with state-of-the-art cybersecurity datasets such as NSL-KDD and UNSW-NB15 showed that the proposed framework performed better than traditional unsupervised methods like One-Class SVM, Isolation Forest, K-Means and standalone Autoencer. The framework demonstrated better accuracy, precision, recall, F1-score and ROC-AUC scores and low false positive rates and high classification stability. Statistical validation of 10-fold cross-validation was also used to validate the strength and ability of the proposed model to generalize. The key finding of this work is a scalable hybrid framework of anomaly detection using clustering-assisted learning of deep representations that can readily process high-dimensional streaming data. Future solutions will include adaptive optimization of thresholds, attention-based deep learning models, federated anomaly detection systems, and edge deployment in real-time to support large scale intelligent monitoring settings.

References

1. Aggarwal, C. C., & Yu, P. S. (2001). Outlier detection for high dimensional data. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 37–46. https://doi.org/10.1145/375663.375668
2. Alghushairy, O., Alsini, R., Soule, T., & Ma, X. (2020). A review of local outlier factor algorithms for outlier detection in big data streams. *Big Data and Cognitive Computing, 5*(1), 1. https://doi.org/10.3390/bdcc5010001
3. Breunig, M. M., Kriegel, H. P., Ng, R. T., & Sander, J. (2000). LOF: Identifying density-based local outliers. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 93–104. https://doi.org/10.1145/342009.335388
4. Chalapathy, R., & Chawla, S. (2019). Deep learning for anomaly detection: A survey. *ACM Computing Surveys, 54*(2), 1–38. https://doi.org/10.1145/3439950
5. Han, D., Wang, Z., Chen, W., Zhong, Y., Wang, S., Zhang, H., Yang, J., Shi, X., & Yin, X. (2021). DeepAID: Interpreting and improving deep learning-based anomaly detection in security applications. *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 3197–3217. https://doi.org/10.1145/3460120.3485374
6. Kumar, A., Gupta, M., & Rajput, R. S. (2021). A survey on high-dimensional data analysis: Challenges and methods. *International Journal of Computer Science, 5*(3), 87–101.
7. Liu, F. T., Ting, K. M., & Zhou, Z. H. (2012). Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data, 6*(1), 1–39. https://doi.org/10.1145/2133360.2133363
8. Liu, H., Zhao, B., Guo, J., Zhang, K., & Liu, P. (2024). A lightweight unsupervised adversarial detector based on autoencoder and isolation forest. *Pattern Recognition, 147*, 110127. https://doi.org/10.1016/j.patcog.2023.110127
9. Liu, Y., Garg, S., Nie, J., Zhang, Y., Xiong, Z., Kang, J., & Hossain, M. S. (2020). Deep anomaly detection for time-series data in industrial IoT: A communication-efficient on-device federated learning approach. *IEEE Internet of Things Journal, 8*(8), 6348–6358. https://doi.org/10.1109/JIOT.2020.3031801

10. Pang, G., Shen, C., Cao, L., & van den Hengel, A. (2021). Deep learning for anomaly detection: A review. *ACM Computing Surveys, 54*(2), 1–38. https://doi.org/10.1145/3439950
11. Qiao, Y., Wu, K., & Jin, P. (2021). Efficient anomaly detection for high-dimensional sensing data with one-class support vector machine. *IEEE Transactions on Knowledge and Data Engineering, 35*(1), 404–417. https://doi.org/10.1109/TKDE.2021.3058105
12. Rousseeuw, P. J., & Hubert, M. (2018). Anomaly detection by robust statistics. *WIREs Data Mining and Knowledge Discovery, 8*(2), e1236. https://doi.org/10.1002/widm.1236
13. Steinbach, M., Ertöz, L., & Kumar, V. (2004). The challenges of clustering high dimensional data. In M. J. Zaki & C. C. Aggarwal (Eds.), *New directions in statistical physics* (pp. 273–309). Springer. https://doi.org/10.1007/978-3-662-08968-2_16
14. Thudumu, S., Branch, P., Jin, J., & Singh, J. (2020). A comprehensive survey of anomaly detection techniques for high dimensional big data. *Journal of Big Data, 7*(1), 42. https://doi.org/10.1186/s40537-020-00320-x
15. Wang, M., Li, D., & Zhang, X. (2020). Clustering high-dimensional data: A review of recent advances. *IEEE Transactions on Knowledge and Data Engineering, 32*(4), 934–948. https://doi.org/10.1109/TKDE.2019.2892315
16. Zhang, R., Zhang, S., Lan, Y., & Jiang, J. (2008). Network anomaly detection using one-class support vector machine. *Proceedings of the International MultiConference of Engineers and Computer Scientists*, 1, 1–6.
17. Zong, B., Song, Q., Min, M. R., Cheng, W., Lumezanu, C., Cho, D., & Chen, H. (2018). Deep autoencoding Gaussian mixture model for unsupervised anomaly detection. *International Conference on Learning Representations*, 1–19.
18. Siaffa Wright. (2026). AI-Assisted Adaptive Impedance Matching Network for Wideband IoT RF Front-Ends. *National Journal of RF Circuits and Wireless Systems* , *3*(3), 9-17.
19. Moti Ranjan Tandi. (2026). Stochastic Wind Field Learning Using Multi-Fidelity Surrogate Models for Robust Micro-Siting Optimization. *Journal of Scalable Data Engineering and Intelligent Computing*, 1-8.
20. Saravanakumar Veerappan. (2025). Interference-Aware Learning Control Mechanisms for Electromagnetically Coupled Drive Systems. *Journal of Wireless Intelligence and Spectrum Engineering*, 1–9.