Comparative Analysis of Dimensionality Reduction Techniques for Cybersecurity in the SWaT Dataset

Mehmet Bozdal
Abdullah Gul University
Kadir Ileri (kileri@bandirma.edu.tr)
Bandirma Onyedi Eylul University
Ali Ozhakraman
Istanbul Technical University

Research Article

Keywords: intrusion detection, secure water treatment dataset, convolutional neural networks, dimensionality reduction, gated recurrent unit

Posted Date: May 10th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-2904250/v1

Additional Declarations: No competing interests reported.

Comparative Analysis of Dimensionality Reduction Techniques for Cybersecurity in the SWaT Dataset

First Mehmet Bozdal[1†], Second Kadir Ileri[2*†], Third Ali Ozkahraman[3†]

[1]Electrical & Electronics Engineering, Abdullah Gul University, Kayseri, 38000, Turkey.
[2*]Electrical & Electronics Engineering, Bandirma Onyedi Eylul University, Balikesir, 10200, Turkey.
[3]Electrical & Electronics Engineering, Istanbul Technical University, Istanbul, 34000, Turkey.

*Corresponding author(s). E-mail(s): kileri@bandirma.edu.tr;
Contributing authors: mehmet.bozdal@agu.edu.tr;
ozkahraman@itu.edu.tr;
†These authors contributed equally to this work.

Abstract

The Internet of Things (IoT) has revolutionized the functionality and efficiency of distributed cyber-physical systems, such as city-wide water treatment systems. However, with increased connectivity comes the risk of cybersecurity threats. In this research, we propose an Intrusion Detection System (IDS) for securing the Secure Water Treatment (SWaT) dataset using a 1D Convolutional Neural Network (CNN) model enhanced with a Gated Recurrent Unit (GRU). The proposed method outperforms existing methods by achieving 99.68% accuracy and F1 score of 0.9869. The paper also explores dimensionality reduction methods, including Autoencoders, Generalized Eigenvalue Decomposition (GED), and Principal Component Analysis (PCA). The research findings highlight the importance of balancing dimensionality reduction with the need for accurate intrusion detection. It is found that PCA provided better performance compared to the other techniques, as reducing the input dimension by 90.2% resulted in only a 2.8% and 2.6% decrease in the accuracy and F1 score, respectively.

Keywords: intrusion detection, secure water treatment dataset, convolutional neural networks, dimensionality reduction, gated recurrent unit

1 Introduction

The Internet of Things (IoT) is a technology that enables the connection of every-day devices, such as appliances, vehicles, and industrial equipment, to the internet. This allows these devices to communicate with one another and with other systems, and to be controlled and monitored remotely. The increased connectivity provided by IoT has had a significant impact on industrial control systems, which were previously closed off from the outside world.

In the past, industrial control systems (ICS) were primarily used to control and monitor industrial processes within a single facility or on a small scale. With the advent of IoT, however, these systems can now be connected to the internet, enabling remote monitoring and control. This allows for city-wide or nation-wide distributed systems to work collaboratively and efficiently, with the ability to share information and coordinate actions across different locations.

Although connectivity has many benefits, it also brings the danger of cyberattacks. An attacker can access the communication channel and control the system and implement an attack. The attack may have various effects from simply unavailability of service to catastrophic system failure. As industrial control systems become connected to the internet, they become more vulnerable to cyberattacks.

There are examples of cyberattacks targeting industrial control systems (ICS) in recent years. In 2000, a former employee maliciously commanded SCADA (Supervisory Control and Data Acquisition) radio-controlled sewage [1]. He caused hundreds of thousands of raw sewages to spill out around various parts of the city in Australia.

One of the most well-known examples of an ICS cyberattack is the Stuxnet worm [2], which was discovered in 2010. The worm specifically targeted the software used to control industrial processes at an Iranian nuclear facility. The attack caused physical damage to the centrifuges used to enrich uranium, setting back the facility's operations.

In 2015 a malicious cyberattacks were targeted the Ukraine power grid [3], causing widespread power outages across the country. The attackers used spear-phishing emails to gain access to the network, and then used malware to disrupt the operations of the power plants.

WannaCry ransomware attack affected thousands of computers including industrial control plants [4]. Triton malware [5], which specifically targeted the industrial control systems used to operate critical infrastructure, was discovered in 2017. The malware manipulated the Triconex Safety Instrumented System (SIS) controllers, which are used to monitor and control industrial processes in facilities such as oil refineries and chemical plants. Although the full extent of damage caused by these attacks are not publicized, the attacks demonstrate the potential consequences of a successful cyber-attack on industrial control systems, which can include physical damage, disruption of operations, and even fatalities.

Attacks on ICS can range from simple disruptions of service to catastrophic failures that can have major physical consequences. Given the potential consequences of a successful attack, it is important to take the necessary steps to protect industrial control systems especially critical infrastructure. Therefore, organizations that deploy IoT-enabled industrial control systems need to be aware of these security risks and take appropriate measures to protect against them. This includes implementing robust security protocols, monitoring for and responding to potential security threats, and providing employee education and training to raise awareness of security risks. One common practice of protecting ICS is the use of intrusion detection systems (IDS). Re-searchers have proposed various IDSs to identify and detect intrusions and help secure cyber-physical systems. However, the efficiency and effectiveness of IDSs can be improved through feature selection and feature reduction algorithms.

In this research, we propose a method for securing the Secure Water Treatment (SWaT) dataset by implementing an IDS using a one-dimensional Convolutional Neural Network (CNN) model enhanced with a Gated Recurrent Unit (GRU). Additionally, we explore various dimensionality reduction techniques, such as autoencoders, Generalized Eigenvalue Decomposition (GED), and Principal Component Analysis (PCA). The goal of the paper is to determine the optimal feature subset that can improve the efficiency of the model without compromising its accuracy. In light of above, the contributions of the paper are as follows:

- A novel IDS approach based on a 1D CNN model enhanced with a GRU is achieved for securing the SWaT dataset, which outperforms traditional IDS methods in terms of accuracy and robustness.
- The impacts of dimensionality reduction techniques are evaluated.
- Insights have been provided into the trade-off between feature reduction and detection accuracy.
- The importance of balancing feature reduction and detection accuracy is demonstrated for effective intrusion detection in cyber-physical systems.

The rest of this paper is organized as follows. Section 2 gives brief background information of SWaT dataset, GRU, and CNN along with the related works. The proposed method and implementation details are explained in Section 3. The experimental result and discussion are presented in section 4 followed by a conclusion in Section 5.

2 Material & Methods

2.1 Secure Water Treatment (SWaT) dataset

The Secure Water Treatment (SWaT) dataset [6], which is widely used as a testbed for water treatment, is used in this experiment. The SWaT system produces filtered water at a rate of 5 gallons per hour and was designed under the supervision of Singapore's Public Utility Board. It contains six stages labelled as P1 through P6 as shown in Figure 1. Each stage is operated by a PLC (e.g. PLC 1 controls stage P1) using a distributed control strategy.

The system can be divided into two levels: Level 0 and Level 1. At Level 0, Programmable Logic Controllers (PLCs) acquire data from local sensors such as an acidity

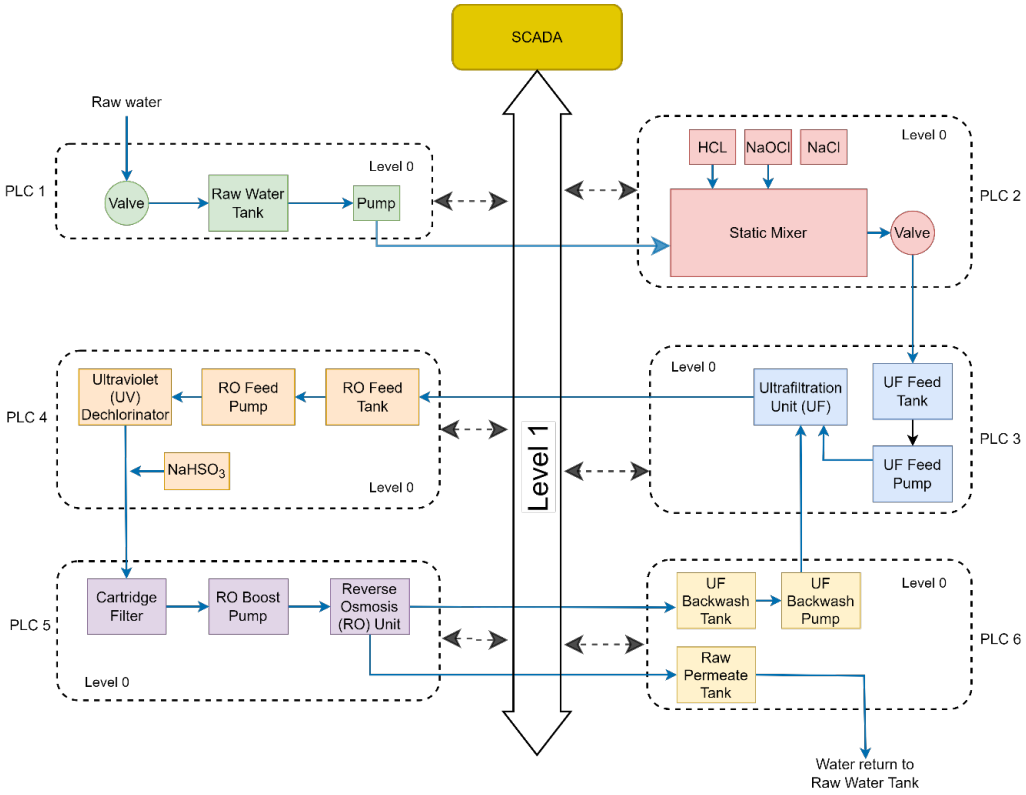

Fig. 1 The process flowchart of SWaT system.

analyzer, water level sensor, and flow meter, and handle the actuators such as valves and pumps. At Level 1, the PLCs communicate with each other using a separate network, which connects all six stages to the Supervisory Control and Data Acquisition (SCADA) system.

PLC 1 controls the flow of raw water by opening or closing the valves connected to the inlet and outlet of the raw water tank in Stage 1. After chemical dosing in Stage 2, the water is fed to Stage 3 for the Ultra Filtration (UF). From there, the UF feed pump forwards the water to the Reverse Osmosis (RO) feed tank in Stage 4. Before entering the RO process, the water passes through an ultraviolet (UV) de-chlorinator to remove any free chlorine. In Stage 5, the RO process removes inorganic impurities from the de-chlorinated water. The filtered water produced by the RO process is stored in the permeating tank in Stage 6 for distribution, and Stage 6 also handles the cleaning of the UF membranes through the backwash process.

The dataset comprises 11 days of total attack duration, with the first seven days being attack-free. It contains 946,722 samples and 51 attributes. Attacks were implemented in Level 1, where Programmable Logic Controllers (PLCs) communicate with the SCADA system. In this level, the data packets are manipulated, and malicious

messages are transmitted to the SCADA system. The attack duration varied from a few minutes to a few hours.

2.2 Related Work

Rule-based anomaly detection is a widely used method for identifying unusual activity in a system based on predefined rules. These rules can be based on patterns and characteristics of known malicious activity, and if a known pattern is observed, it is considered an anomaly. The rules can also be based on the normal behavior of the system, such as setting threshold values for specific parameters. If these values are exceeded or not met, an alarm is triggered.

Adepu and Mathur [7] proposed a novel method for distributed attack detection by utilizing process invariants derived from Piping and Instrumentation Diagrams (P&IDs) based on physical properties of the system. The authors applied this method to a SWaT system, which has chemical processes as well, but due to the nonlinearity of these processes, only physical invariants were used. Although the method does not produce any false alarms, it fails to identify some attack types like denial of service. Furthermore, the process of deriving the invariants is currently a manual process, which may limit the scalability of the method. Future research should focus on automating this process to improve the overall performance of the method.

Another example of rule-based anomaly detection is Logical Analysis of Data (LAD) which was implemented by Das et al. [8]. This method allows for near-real-time processing with low computational power, making it an efficient and cost-effective way to detect some types of cyberattacks. However, it is important to note that reliance on predefined rules alone can be circumvented [9], highlighting the need to supplement rule-based methods with other security measures such as behavioral analysis and machine learning. Al-Dhaheri et. al [10] proposed hybrid intrusion detection system. Rule-based IDS that checks limits and safety values, model-based monitoring that implements physical model, and data-driven approach for non-linear modelling.

Aboah et al. [11] proposed a neural network with a one-class objective function (NN-One-class) which improves the detection performance compared to some of the previous methods. However, the training time can be quite extensive, taking up to 110 minutes with an NVIDIA Tesla T4 GPU and a RAM of 32GB. Given the complexity of the data, this represents a significant amount of resources.

Kravchik and Shabtai [12] proposed a 1D Convolutional Neural Network (CNN) to identify cyberattacks on the SWaT dataset. They implemented dedicated anomaly detectors for each stage of the SWaT system to improve the performance. The results showed that independent analysis of each stage outperforms a single model for the whole system. However, as the stages of the SWaT system are dependent on each other, it is important to also investigate the inter-stage dependencies in order to further improve the performance of the detection system.

Zhou et. al. [13] suggested to use temporal and spatial correlation as temporal correlation alone is not beneficial for high dimensional data. They have implemented Graph Attention Network (GAT) with Multihead Dynamic Attention (MDA). The implementation leverages of relationship between various sensors thanks to MDA.

Nedeljkovic and Jakovljevic [14] implemented semi-supervised IDS by using CNN-based auto regression. They applied Finite Impulse Response (FIR) filter to remove high frequency noise.

Dillon et. al [15] showed that design knowledge increases the efficiency of the IDS. One reason behind that is when data consist of binary values and analogue ones, machine learning algorithms can be biased towards binary ones and ignore them. Experimental results show a 5% increase in the detection by using design knowledge.

There are also research papers that implement dimension reduction. Li et. al. [16] proposed a method called "end-to-end anomaly detection" for detecting anomalies using a digital twin. The proposed method uses a multidimensional deconvolutional network and attention mechanism with PCA to detect anomalies quickly in real-time. However, the performance of the method, F1=0.94, is not acceptable for critical infrastructure. Alimi et al. [17] applied PCA to various supervised learning algorithms. They achieved the best performance for the SWaT dataset with the J48 decision tree classifier, however, the F1 score was 0.814. Priyanga et al. [18] proposed a hyper-graph-based anomaly detection technique. The proposed algorithm involves two phases: dimensionality reduction using enhanced principal component analysis (EPCA) and anomaly detection with HG-based convolution neural network (CNN). El-Nour et al. proposed framework [19] involving two isolation forest models and PCA. Although these methods implemented dimensionality reduction algorithms, they do not emphasize on dimensionality reduction. This article explores the limitations of current dimensionality reduction methods and discusses the importance of dimensionality reduction.

2.3 Convolutional Neural Network and Gated Recurrent Unit

The SWaT dataset contains time-series data that represents the state of an industrial control system at different points in time. To analyze this type of data, a 1D Convolutional Neural Network (CNN) can be used to identify patterns and anomalies in the data. Additionally, a Gated Recurrent Unit (GRU) can be used to capture the temporal dependencies between the sensor readings. By combining these two techniques, a more powerful hybrid model can be developed that captures both the local features and the long-term dependencies of the data. This can lead to better detection of anomalies and ultimately improve reliability and security.

2.4 Dimensionality Reduction Techniques

In machine learning, dimensionality reduction is a common technique used to reduce the number of features in a dataset. Feature reduction techniques can help to reduce the complexity of a dataset, remove noise, and improve the efficiency of machine learning algorithms. We have explored the most common dimensionality reduction techniques including Principal Component Analysis (PCA), Generalized Eigenvalue Decomposition (GED), and Autoencoders.

2.4.1 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a commonly used statistical technique for dimensionality reduction in data analysis and machine learning. The main goal of PCA is to identify patterns and structure in high-dimensional data by reducing the number of variables and retaining the most important information.

Once the principal components are identified, data can be projected onto a lower-dimensional subspace by selecting a subset of the principal components. This new subspace retains the most important information from the original high-dimensional dataset while reducing the number of variables.

2.4.2 Generalized Eigenvalue Decomposition (GED)

Generalized Eigenvalue Decomposition (GED) is a dimension reduction technique that is used to reduce the dimensionality of high-dimensional data while preserving the information contained in the original data. GED finds a linear transformation of the original data that maximizes the ratio of between-class variance to within-class variance.

2.4.3 Autoencoders

An autoencoder is a neural network that can be used for dimensionality reduction by compressing high-dimensional data into a lower-dimensional latent representation as shown in Figure 2. The autoencoder consists of an encoder that maps the input data to the latent layer, a bottleneck layer that represents the compressed data, and a decoder that reconstructs the original data from the latent representation. By training the network to minimize the difference between the input and reconstructed output, the network learns to identify the most important features in the data and discard the less important ones.

The size of the latent space is an important consideration when designing an autoencoder, as it determines how much information will be retained after compression. If the latent space is too small, the autoencoder may lose important information and result in the poor reconstruction of the input data. On the other hand, if the latent space is too large, the autoencoder may overfit and memorize the training data, resulting in poor generalization to new data.

3 Proposed Method and Experiment

3.1 Proposed Method

In this research, a novel approach has been employed to enhance the performance of the one-dimensional convolutional neural network (1D CNN) combined with GRU. The proposed approach aims to enhance performance by utilizing the strengths of both 1D CNN and GRU. This integration allows for the learning of spatial and temporal features of input data, facilitating the capturing of complex patterns in time-series data.

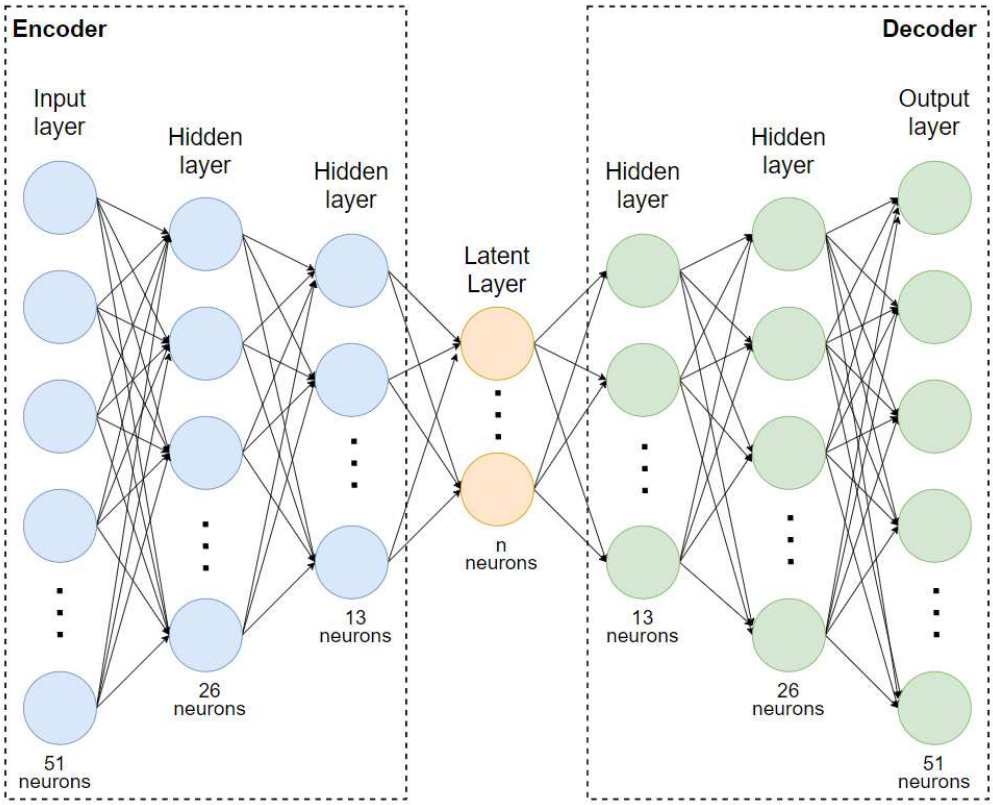

Fig. 2 The architecture of the autoencoder used in this experiment.

The proposed 1D CNN model consists of three convolutional layers, each with a filter size of 16 and a kernel size of 5, 3, and 2 respectively as shown in Figure 3a. The filter size refers to the number of filters or feature maps used in each convolutional layer. The kernel size, on the other hand, refers to the size of the convolutional filter or window that is moved across the input sequence to extract features. A smaller kernel size allows the network to capture more local features, while a larger kernel size captures more global features.

After the three convolutional layers, the output feature maps from each layer are concatenated into a single tensor. The purpose of this is to combine the features learned at different levels of abstraction into a single representation.

The concatenated tensor is then fed into a max pooling layer, which reduces the spatial dimensions of the tensor by taking the maximum value within a specified window. This helps to extract the most salient features from the input sequence while reducing the computational cost of the network.

Batch normalization is then applied to normalize the output of the previous layer, which helps to speed up training and improve the generalization of the model. Finally,

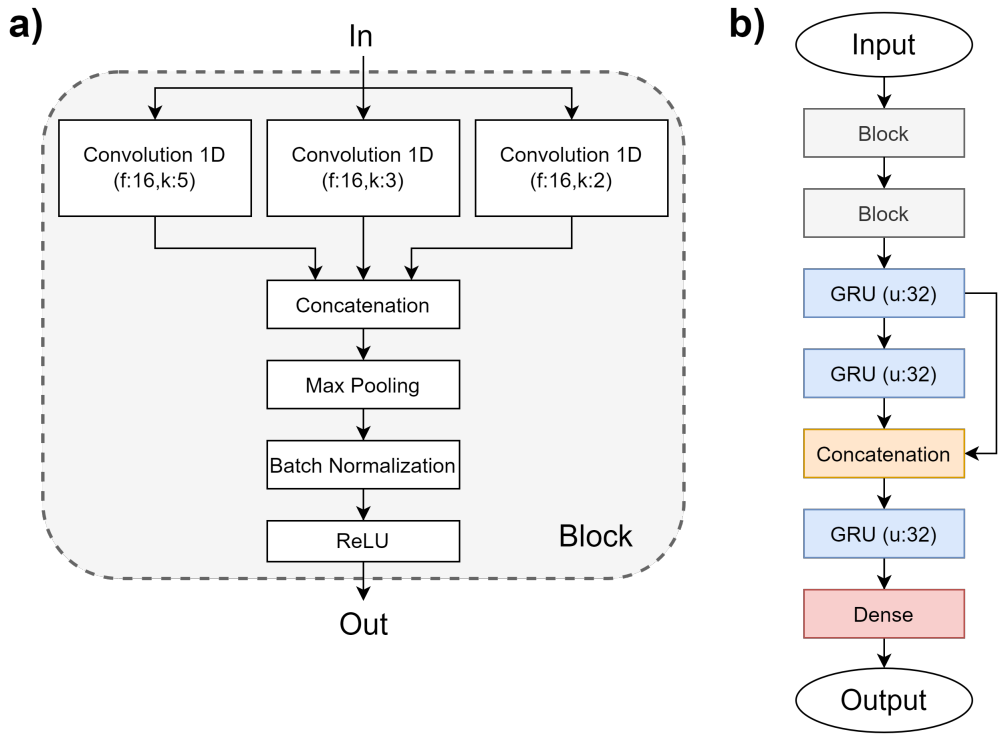

Fig. 3 a) The architecture 1D CNN block used in the experiment along with b) the whole architecture including GRU layers.

the ReLU activation function is applied elementwise to the output of the batch normalization layer, which introduces non-linearity to the model and helps to extract more complex features.

As shown in Figure 3b, after the two blocks of the CNN, the output is fed into GRU layers, which can capture longer-term dependencies in the input sequence. Finally, the output of the GRU layer is passed through a dense layer, which produces the final output of the model.

3.2 Parameter Selection and Experimental Setup

In order to achieve optimal performance of the proposed method, it is important to carefully tune its hyperparameters. Hyperparameters are settings that are not learned during training but are set before training and can have a significant impact on the performance of the model.

The number of epochs is an important hyperparameter that determines the number of times the entire dataset is used to train the model. In this research, we conducted an epoch analysis to determine the optimal number of epochs for training the 1D

CNN-GRU model on the SWaT dataset. We trained the model for different numbers of epochs ranging from 1 to 100 and evaluated its performance.

Figure 4 illustrates the trends of validation loss and accuracy as a function of the number of epochs. The optimal epoch number is located at epoch number 10, where the minimum validation loss and maximum validation accuracy intersect.

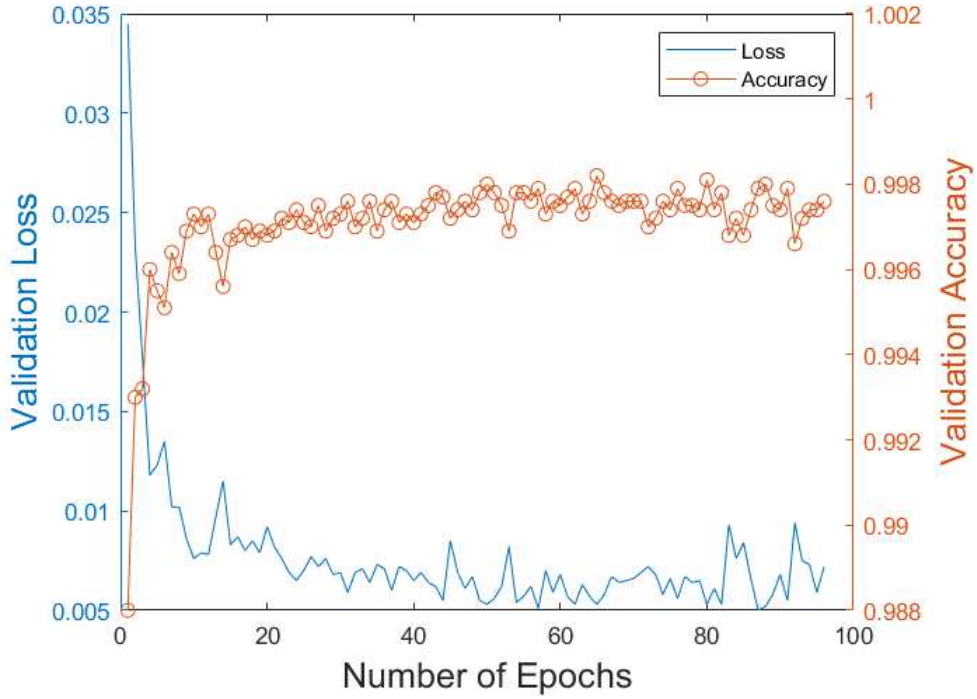

Fig. 4 Determining the optimal number of epochs for model training using validation loss and accuracy.

Other hyperparameters such as batch size, learning rate, and optimizer can significantly impact the performance of a deep-learning model. Therefore, it is important to optimize these hyperparameters to achieve the best possible performance. In this study, we chose a batch size of 32, a learning rate of 0.001, and the Adam optimizer based on their effectiveness in previous studies and our experimentation on the SWaT dataset.

4 Results and Discussion

The proposed method is compared with the state-of-the-art techniques on the SWaT dataset to demonstrate its effectiveness. Additionally, the results of our dimensionality reduction analysis using PCA, GED, and autoencoders are presented to show the impact of feature reduction on the performance of the intrusion detection system.

4.1 Evaluation Method

The proper testing of an Intrusion Detection System (IDS) is a crucial step in evaluating its effectiveness. To ensure the accuracy and reliability of the proposed IDS model, we conducted a comprehensive analysis of its performance.

Our proposed model employs a binary classifier to differentiate between authentic messages and potential attacks. As a result, there are four possible outcomes: false negative, false positive, true negative, and true positive. A true positive occurs when an attack is correctly identified by the system, while a true negative occurs when an authentic message is correctly accepted as such. In contrast, a false positive occurs when an authentic message is labeled as an attack, and a false negative occurs when an attack is labeled as an authentic message.

To assess the performance of our proposed IDS model, we calculated the values for FN, FP, TN, and TP. These values provide important insights into the system's ac-curacy and effectiveness in detecting potential attacks. Additionally, we calculated several key metrics such as accuracy, precision, and recall values.

The accuracy metric evaluates the percentage of correct predictions made by the model, whereas the precision metric assesses the percentage of true positives among all positive predictions. Recall metric evaluates the percentage of true positives detected by the system among all actual attacks. By considering all of these metrics, we can assess the overall performance of the IDS model and determine its efficiency in detecting potential attacks.

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \tag{1}$$

Precision is a performance metric used in evaluating the effectiveness of an Intrusion Detection System (IDS). Specifically, precision evaluates the percentage of true positive predictions made by the system out of all positive predictions. It provides an important measure of the system's ability to accurately identify potential attacks while minimizing the number of false positives.

$$Precision = TP/(TP + FN) \tag{2}$$

Recall (also known as sensitivity or detection rate) is a performance metric used in evaluating the effectiveness of an IDS. Specifically, recall measures the percentage of true positive predictions made by the system out of all actual positive cases.

$$Sensitivity(Recall) = TP/(TP + FN) \tag{3}$$

The F1 score is defined as the harmonic mean of recall and precision, where a higher score indicates better performance. By taking the harmonic mean, the F1 score places more emphasis on the lower of the two metrics, meaning that a model with high precision but low recall (or vice versa) will have a lower F1 score than a model with both high recall and high precision.

$$F1 = (2 * (Precision * Recall))/(Precision + Recall) \tag{4}$$

4.2 Comparison with the State-of-the-art Techniques

As many researchers use this dataset, it serves as a common benchmark for evaluating and comparing the performance of different methods. The proposed method is compared with other state-of-the-art methods.

Table 1 presents the performance metrics for the proposed method along with other State-of-the art proposals. There is a trade-off between Precision and Recall. These two metrics measure different aspects of a classifier's performance, and optimizing one metric often comes at the expense of the other.

Table 1 The comparison of methods that use the SWaT dataset.

Reference	Accuracy	F1	Precision	Recall
CNN-GRU-SDA [20]	-	0.91	0.99	0.85
CNN-FIR [14]	97.846	0.902	0.988	0.830
1D CNN [12]	97.195	0.871	0.968	0.791
NN-PCA [21]	97.408	0.885	0.911	0.860
Monitoring System [10]	-	0.925	**1**	0.861
STAE-AD [22]	-	0.880	0.960	0.815
NN-one class [11]	-	0.870	0.940	0.820
EPCA-HG-CNN [18]	98.02	0.9805	0.9771	0.9839
Digital-twin [16]	-	90.59	0.923	0.961
DIF [19]	97.375	0.882	0.935	0.835
1D CNN-GRU (This Paper)	**0.9968**	**0.9869**	0.9855	**0.9882**

*Best results are bold.

The Table 1 depicts that some models achieved high Precision scores but lower Recall scores (e.g., CNN–FIR), while others achieved higher Recall scores but lower Preci-sion scores (e.g., EPCA-HG-CNN). The proposed CNN-GRU model achieved the best overall performance, achieving an impressive accuracy score of 0.9968, F1 score of 0.9869, precision of 0.9855, and recall of 0.9882.

4.3 Analysis of Dimensionality Reduction Techniques

Dimensionality reduction is a commonly used technique in machine learning for reducing the number of features in a dataset. It helps in reducing the complexity of the dataset, removes noise, and improves efficiency. In this research, we explored the effectiveness of three commonly used dimensionality reduction techniques: Generalized Eigenvalue Decomposition (GED), Autoencoders, and Principal Component Analysis (PCA) for improving the performance of the proposed IDS.

4.3.1 Generalized Eigenvalue Decomposition

The magnitude of the eigenvalues obtained through GED can provide important information about the quality of the dimensionality reduction. Figure 5 presents the Magnitudes of eigenvalues for eigenvectors.

Table 2 presents experimental results for GED. The accuracy increases from 0.9692 for 5 eigenvectors to 0.9950 for 25 eigenvectors. Similarly, the F1 score consistently increases from 0.8681 to 0.9793. The precision of the IDS also increases as the number

of eigenvectors increases, with the highest precision of 0.9843 achieved with 20 eigenvectors. The recall of the IDS is highest for 25 eigenvectors with a value of 0.9781, indicating that the IDS with 25 eigenvectors is better at detecting true positive cases. The true positive (TP) values increase with the number of eigenvectors, while the false negative (FN) values decrease, indicating that the IDS is more capable of detecting true positive cases with a higher number of eigenvectors. However, the false positive (FP) values slightly increase as the number of eigenvectors increases, which suggests that increasing the number of eigenvectors may result in a higher rate of false alarms.

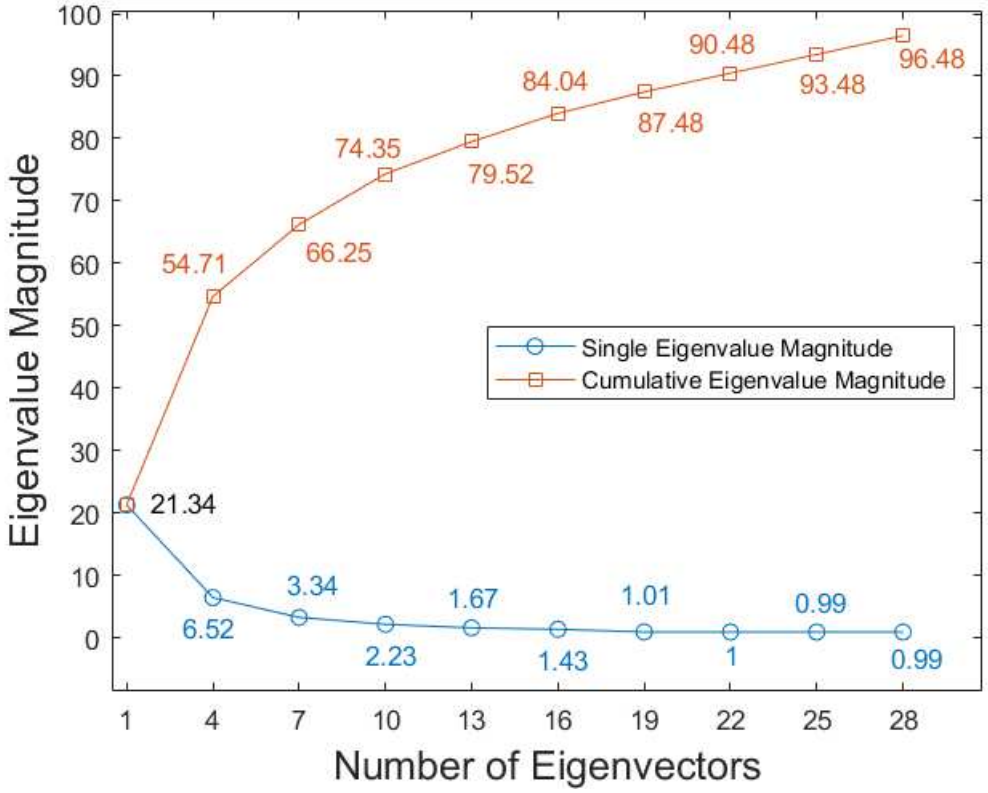

Fig. 5 The magnitudes of eigenvalues for eigenvectors.

4.3.2 Autoencoder

Choosing the appropriate number of latent layers can be a challenging task, and it often requires experimentation and tuning to find the optimal number for a given problem. Typically, the number of latent layers is determined by balancing the trade-off between model complexity and performance on the validation set.

Table 2 The performance analysis of generalized eigenvalue decomposition.

# of Eigenvectors	Accuracy	F1	Precision	Recall	TP	TN	FP	FN
5	0.9692	0.8681	0.8993	0.8391	9122	78092	1021	1749
10	0.9927	0.9700	0.9687	0.9713	10559	78772	341	312
15	0.9947	0.9781	0.9834	0.9727	10575	78935	178	296
20	0.9949	0.9789	0.9843	0.9735	10583	78945	168	288
25	0.9950	0.9793	0.9805	0.9781	10633	78901	212	238

Table 3 presents the performance analysis of an autoencoder with different numbers of latent layers (n). The accuracy of the model increases with the number of latent layers, reaching its highest value of 0.9965 with 25 latent layers. Similarly, the F1 score, precision, and recall increase with the number of latent layers, with the highest values being 0.9855, 0.9823, and 0.9881, respectively, for 25 latent layers.

Table 3 The performance analysis of autoencoder.

# of Latent Layer (n)	Accuracy	F1	Precision	Recall	TP	TN	FP	FN
5	0.9849	0.9479	0.9868	0.8864	9637	78984	129	1234
10	0.9932	0.9714	0.9918	0.9518	10347	79028	85	524
15	0.9926	0.9685	0.9926	0.9455	10279	79036	77	592
20	0.9921	0.9665	0.9887	0.9452	10275	78996	117	596
25	0.9965	0.9855	0.9823	0.9881	10742	78925	188	129

4.3.3 Principal Component Analysis

PCA aims to retain the most important information from the original high-dimensional dataset while reducing the number of variables. The number of principal components that should be retained depends on the amount of variance they explain. Figure 6 shows the variance of each principal component and accumulated one. It is observed that the first principal component explains the most variance, followed by the second and third principal components. As more principal components are added, the amount of explained variance gradually decreases. In this specific case, it seems that retaining the first 5 principal components can capture a significant amount of the variation in the data, as they explain over 99.5% of the variance.

Table 4 depicts the performance analysis of the proposed method using PCA for different numbers of components. The result shows that the performance of the intrusion detection model does not degrade significantly even when the number of principal components is reduced by 90.2%. Specifically, when the number of principal components is reduced to 5, the model achieves an accuracy of 0.9909 and an F1 score of 0.9613. When the number of principal components is increased to 20, the model achieves an accuracy of 0.9969 and an F1 score of 0.9873.

The analysis reveals an interesting trend regarding the trade-off between true positive and false negative values. As the number of components increases, the true positive values consistently increase while the false negative values decrease. This finding suggests that the proposed method becomes more capable of correctly detecting positive

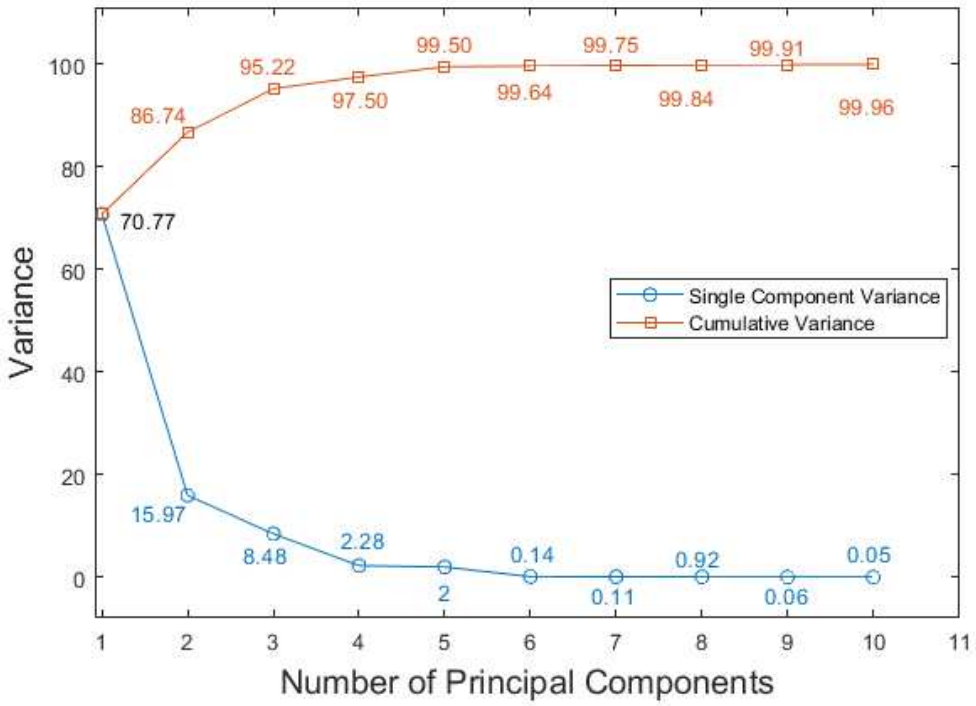

Fig. 6 The variance of PCA components.

Table 4 The performance analysis of PCA.

# of Component	Accuracy	F1	Precision	Recall	TP	TN	FP	FN
5	0.9909	0.9613	0.9892	0.9351	10165	79002	111	706
10	0.9967	0.9863	0.9857	0.9869	10729	78957	156	142
15	0.9967	0.9862	0.9839	0.9886	10747	78937	176	124
20	0.9969	0.9873	0.9832	0.9915	10778	78929	184	93
25	0.9961	0.9839	0.9781	0.9898	10760	78873	240	111

cases as the number of components increases. In contrast, the false positive values remain relatively low across all component numbers, indicating that the pro-posed method can maintain a low rate of false alarms even with an increased number of components.

4.4 Discussion and Future Work

Our findings, summarized in Table 5, suggest that carefully balancing dimensionality reduction with the need for accurate intrusion detection is critical for achieving optimal performance. It is found that PCA was the most effective dimension reduction

technique among the three methods evaluated, as it resulted in the best balance between number of dimension and accuracy. PCA can slightly improve accuracy and F1 score of CNN-GRU architecture with 20 components. On the other hand, reducing the input features by 90.2% using PCA resulted in only a 2.6% decrease in the F1 score of the intrusion detection system. When the number of components is decreased, pure CNN-GRU model outperforms all experimented dimensionality reduction methods. This suggests that there may be trade-offs between reducing dimensionality and maintaining accuracy, and that each situation may require a different approach depending on the specific goals and constraints of the system being used. Overall, the findings suggest that careful consideration and testing of different dimensionality reduction techniques is necessary for optimization.

Table 5 The comparison of the dimensionality reduction techniques with 1D CNN-GRU.

Method	Accuracy	F1	Precision	Recall	TP	TN	FP	FN
1D CNN-GRU	0.9968	0.9869	0.9855	0.9882	10743	78955	158	128
Autoencoder	0.9965	0.9855	0.9823	0.9881	10742	78925	188	129
GED	0.9950	0.9793	0.9805	0.9781	10633	78901	212	238
PCA	0.9969	0.9873	0.9832	0.9915	10778	78929	184	93

Although successful results are observed, there is some room to improve the current system. The method was evaluated offline. Future work could explore the feasibility of implementing the proposed method in real time to provide continuous monitoring and early detection of potential cyberattacks on critical infrastructure systems.

While this research focused on the SWaT dataset, future work could explore the effectiveness of the proposed method on other datasets related to critical infrastructure, such as power grids or transportation systems. This would provide insights into the generalizability of the proposed method.

5 Conclusion

This research investigates the use of a 1D CNN and GRU model on the SWaT dataset. The main aim of the research is to improve the performance of the model by utilizing the complementary strengths of both models. The results demonstrate that combining the 1D CNN and GRU models can significantly enhance the accuracy of the model. Moreover, the study highlights the importance of dimensionality reduction, indicating that the selection of relevant features can significantly affect the performance of the model.

Declarations

- Funding:The authors declare that no funds, grants, or other support were received during the preparation of this manuscript
- Conflict of interest:The authors have no conflicts of interest to declare. All co-authors have seen and agree with the contents of the manuscript and there is no

financial interest to report. We certify that the submission is original work and is not under review at any other publication.

- Data Availability Statement: SwAT dataset from iTrust Lab is used. Publicly available at: https://itrust.sutd.edu.sg/itrust-labs datasets/#SWaT
- Authors' contributions:All authors contributed to the study conception and design. Material preparation and analysis were performed by Mehmet Bozdal, Kadir Ileri and Ali Ozkahraman. All authors read and approved the final manuscript.

References

[1] Abrams, M., Weiss, J.: Malicious control system cyber security attack case study–maroochy water services, australia. McLean, VA: The MITRE Corporation (2008)

[2] David, K.: The real story of stuxnet. ieee Spectrum **50**(3), 48–53 (2013)

[3] Case, D.U.: Analysis of the cyber attack on the ukrainian power grid. Electricity Information Sharing and Analysis Center (E-ISAC) **388**, 1–29 (2016)

[4] Kovacs, E.: Industrial Systems at Risk of WannaCry Ransomware Attacks. https://www.securityweek.com/industrial-systems-risk-wannacry-ansomware-attacks. Accessed: 2023-01-11

[5] Di Pinto, A., Dragoni, Y., Carcano, A.: Triton: The first ics cyber attack on safety instrument systems. In: Proc. Black Hat USA, vol. 2018, pp. 1–26 (2018)

[6] Laboratory: Secure Water Treatment (SWaT). https://itrust.sutd.edu.sg/itrust-labs_datasets/#SWaT. Accessed: 2023-01-11

[7] Adepu, S., Mathur, A.: Distributed attack detection in a water treatment plant: Method and case study. IEEE Transactions on Dependable and Secure Computing **18**(1), 86–99 (2018)

[8] Das, T.K., Adepu, S., Zhou, J.: Anomaly detection in industrial control systems using logical analysis of data. Computers & Security **96**, 101935 (2020)

[9] Gold, D.: Is Signature- and Rule-Based Intrusion Detection Sufficient? https://www.csoonline.com/article/3181279/is-478signature-and-rule-based-intrusion-detection-sufficient.html. Accessed: 2023-02-28

[10] Al-Dhaheri, M., Zhang, P., Mikhaylenko, D.: Detection of cyber attacks on a water treatment process. IFAC-PapersOnLine **55**(6), 667–672 (2022)

[11] Boateng, E.A., Bruce, J., Talbert, D.A.: Anomaly detection for a water treatment system based on one-class neural network. IEEE Access **10**, 115179–115191 (2022)

[12] Kravchik, M., Shabtai, A.: Detecting cyber attacks in industrial control systems using convolutional neural networks. In: Proceedings of the 2018 Workshop on Cyber-physical Systems Security and Privacy, pp. 72–83 (2018)

[13] Zhou, L., Zeng, Q., Li, B.: Hybrid anomaly detection via multihead dynamic graph attention networks for multivariate time series. IEEE Access **10**, 40967–40978 (2022)

[14] Nedeljkovic, D., Jakovljevic, Z.: Cnn based method for the development of cyber-attacks detection algorithms in industrial control systems. Computers & Security **114**, 102585 (2022)

[15] Cheong Lien Sung, D., MR, G.R., P Mathur, A.: Design-knowledge in learning plant dynamics for detecting process anomalies in water treatment plants (2022)

[16] Li, Z., Duan, M., Xiao, B., Yang, S.: A novel anomaly detection method for digital twin data using deconvolution operation with attention mechanism. IEEE Transactions on Industrial Informatics (2022)

[17] Alimi, O.A., Ouahada, K., Abu-Mahfouz, A.M., Rimer, S., Alimi, K.O.A.: Supervised learning based intrusion detection for scada systems. In: 2022 IEEE Nigeria 4th International Conference on Disruptive Technologies for Sustainable Development (NIGERCON), pp. 1–5 (2022). IEEE

[18] Krithivasan, K., Pravinraj, S., VS, S.S., *et al.*: Detection of cyberattacks in industrial control systems using enhanced principal component analysis and hypergraph-based convolution neural network (epca-hg-cnn). IEEE Transactions on Industry Applications **56**(4), 4394–4404 (2020)

[19] Elnour, M., Meskin, N., Khan, K., Jain, R.: A dual-isolation-forests-based attack detection framework for industrial control systems. IEEE Access **8**, 36639–36651 (2020)

[20] Xie, X., Wang, B., Wan, T., Tang, W.: Multivariate abnormal detection for industrial control systems using 1d cnn and gru. Ieee Access **8**, 88348–88359 (2020)

[21] Kravchik, M., Shabtai, A.: Efficient cyber attacks detection in industrial control systems using lightweight neural networks. arxiv 2019. arXiv preprint arXiv:1907.01216

[22] Macas, M., Wu, C.: An unsupervised framework for anomaly detection in a water treatment system. In: 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), pp. 1298–1305 (2019). IEEE