Enhancing Cyber-Physical System Security: Unsupervised Anomaly Detection in Water Treatment Facilities Using LSTM Networks

Muhammad Mugheera Basharat Mariam Rafique Shreyas Malhotra

May 3, 2024

Abstract

This research focuses on enhancing the reliability and security of safety-critical systems, with a particular emphasis on water treatment systems. Given the critical role these systems play in providing safe drinking water, their vulnerability to cyber and physical threats presents a significant risk of dire consequences, such as environmental pollution and health hazards. This study aims to develop advanced anomaly detection mechanisms tailored specifically for water treatment facilities using data from the Secure Water Treatment (SWaT) testbed. We employ Long Short-Term Memory (LSTM) networks, a type of recurrent neural network ideal for sequential data analysis, to implement an unsupervised learning model. Our methodology includes comprehensive data preprocessing, utilizing techniques such as normalization and feature selection, to prepare the SWaT testbed data for effective model training. This research not only contributes to the theoretical framework of anomaly detection but also provides practical insights for the continuous monitoring and maintenance of safety-critical systems.

1 Introduction

Motivation: In an increasingly interconnected world, safety-critical systems—those whose failure could result in catastrophic outcomes such as loss of life, significant environmental harm, or severe economic repercussions—are becoming indispensable. These systems, including electrical grids and industrial water treatment facilities, form the backbone of all infrastructure. The motivation behind this research is to enhance the reliability of these crucial systems and to ensure their continuous and safe operation against an ever-evolving landscape of cyber and physical threats.

Problem Statement: Our primary focus is the reliability of safety-critical systems, specifically water treatment systems. Water treatment systems are inherently designed to provide clean, safe water to communities. Any disruption or manipulation of these systems through cyber or physical attacks can lead to unsafe drinking water, environmental pollution, or complete operational shutdown. Due to their critical nature, ensuring the resilience of these systems remains a challenge due to the complexity of potential threats they face. This research aims to address the need for advanced detection mechanisms that can identify anomalies and potential threats, thus, preventing failures or attacks.

Importance of Secure Water Treatment Systems: Water treatment facilities, such as those represented by Secure Water Treatment (SWaT) testbed, are complex systems that involve multiple stages of filtration, chemical treatment, and quality assurance to ensure the supply of potable water. These systems' operations rely on the precise management of physical and computational processes, where any anomaly could signal potential threat or system malfunction.

Goal of our project: The primary goal of this research project is to develop predictive and preventative mechanism tailored for water treatment systems and safety-critical systems in general. By leveraging data from the SWaT testbed- a simulated environment that closely imitates real-world operation - we aim to design, test, and validate security protocols that can detect both subtle anomalies and attack attempts effectively.

2 Background and Related Research

2.1 Architecture of LSTM

An LSTM unit consists of a cell (the memory part of LSTM) and three regulators of the cell's state, known as gates. These gates control the flow of information in and out of the cell and are crucial for the LSTM's ability to retain or forget information. These gates are:

1. Forget Gate: This gate decides which information is discarded from the cell state. It looks at the previous hidden state and the current input, and applies a sigmoid function to determine the proportion (between 0 and 1) of each number in the cell state to keep. A value of 0 means "completely forget this" while a value of 1 means "completely retain this." **2. Input Gate:** The input gate has two components: it first applies a sigmoid function to decide which values to update, and a tanh function to create a vector of new candidate values that could be added to the state. The sigmoid output decides which parts of the cell state to update while the tanh output creates a candidate vector of new values. **3. Output Gate:** The output gate decides what the next hidden state should be. It first applies a sigmoid function to the previous hidden state and the current input to decide which parts of the cell state will be output. Then, it applies a tanh function to the cell state (to push the values to be between -1 and 1) and multiplies it by the output of the sigmoid gate, so that only the parts of the cell state that are decided to be output are actually output.

2.2 Cell State

The cell state acts as a transport highway that carries the relevant information through the sequence chain. Information gets added or removed to the cell state via the gates. The cell state is crucial as it can carry relevant information throughout the processing of the sequence, thus alleviating the problem of vanishing gradients by maintaining a more constant error that can be back propagated through time and layers.

2.3 Forward Pass

During the forward pass, the LSTM processes data sequentially, updating its hidden state and cell state at each step of the sequence. The operations at each timestep can be summarized as follows:

1. **Forget Gate Activation**:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{1}$$

   where σ is the sigmoid function, W_f are the weights, b_f is the bias of the forget gate, h_{t-1} is the previous hidden state, and x_t is the input at time t.

2. **Input Gate Activations**:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\tilde{C_t} = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{3}$$

3. **Cell State Update**:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{4}$$

4. **Output Gate Activation**:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{5}$$

5. **Hidden State Update**:

$$h_t = o_t * \tanh(C_t) \tag{6}$$

2.4 Training

LSTMs are trained using backpropagation through time (BPTT), which involves unrolling the network through time and applying the backpropagation algorithm. The training adjusts the weights in the gates to minimize the loss function, typically a function between the predicted output and the actual output.

3 Methodology

This section outlines the methodologies applied to enhance the security of water treatment systems through the development of anomaly detection models. Central to our approach is the preprocessing of data derived from the Secure Water Treatment (SWaT) testbed, the application of machine learning techniques for predictive modeling, and systematic validation of these models against simulated attack scenarios.

3.1 Data Preprocessing

1. Normalization is crucial in preparing data for machine learning models, particularly when variables are different scales and ranges. This disparity can distort performance of the algorithms that are sensitive to feature magnitude, such as gradient based methods. To address this, we apply min-max scaling, which transforms data into a fixed range between 0 and 1, preserving the original distribution without distorting differences in the range of values. This is implemented using the MinMaxScalar from the python library 'pandas.'

2. Handling Missing Values - Data imputation is another crucial step in data preprocessing, given the operational variability in a testbed environment where sensor readings might occasionally fail or transmit erroneous data. We handle missing values in data by applying data imputation techniques that replace these missing data entries with statistically relevant values, which can be the mean or median of the dataset, depending on the distribution and nature of the data.

3. Label Encoding - our dataset includes categorical labels that denote normal operation and various types of attacks on the safety critical system. Since machine learning models inherently require numerical input, we use label encoding to transform these categorical labels into numerical ones. This process is done using the LabelEncode tool from 'pandas' library, which assigns a unique integer to each category of labels.

4. Feature Selection - The selection of appropriate features for training the machine learning models is based on the correlation between different sensor and operational variables gathered from the SWaT testbed. Features that display a high correlation with the target variable - which is indicative of potential predictive power regarding the system's state - are selected for the model training. This step is vital as it directly impacts a model's ability to learn and make accurate predictions.

3.2 Tools and Utilities

Our methodology extensively utilizes python ''pandas' library for data manipulation and preprocessing. This tool allows for effective handling of large datasets, such as those produced by the SWaT testbed, providing functions for scaling, encoding, and inputting data. Additionally, 'pandas' facilitates the extraction and transformation processes, making it a crucial part of our data preprocessing pipeline.

3.3 Unsupervised Anomaly Detection using LSTM

In our approach to enhance security and operational reliability of the Secure Water Treatment (SWaT) system, we employed a sophisticated unsupervised learning model using the Long Short-Term Memory (LSTM) networks. LSTM networks are a type of recurrent neural network which are suited especially for sequential data analysis, making them ideal for time-series data generated by SWaT.

3.3.1 Architecture Overview

The architecture of our LSTM based unsupervised model involves several components and layers designed to process time-series data accurately and effectively:

Input Layer: This layer receives sequences of sensor reading from the SWaT system. These reading include operational metrics including flow rates, pressures, and chemical concentration

LSTM layers: The core of this model which is capable of learning long-term dependencies in data, which is crucial for understanding complex patterns and identifies anomalies that deviate from these patterns.

Output Layer: The output of LSTM layers is used to reconstruct the input sequence. This is a critical part of an unsupervised learning approach where the model learns to accurately reproduce normal operational data.

3.3.2 Algorithmic Approach

The algorithmic approach of our LSTM based anomaly detection model can be described in the following steps:

Preprocessing: Raw data from the SWaT system is first normalized using the standard scalar to ensure consistent scale across all input features. This step is crucial for effective training of neural networks which are sensitive to scale of input data

Data Segmentation: The continuous time-series data is segmented into fixed-size windows where each window of data serves as an input sequence to the LSTM model.

Model Training: The LSTM model is trained on these windows of data in an unsupervised manner. The training objective is to minimize the reconstruction error which is the difference between input sequence and reconstructions.

Anomaly Detection: Once the model is trained, it is used to reconstruct new data sequences. The reconstruction error is computed and if this error exceeds a predefined threshold, the data sequence is flagged as anomalous.

Post-Processing: The deleted anomalies are then analyzed to determine their causes and impacts.

Thus, by utilizing LSTM networks in an unsupervised learning environment, our methodology takes advantage of the patterns in SWaT data to detect anomalies effectively. This approach not only allows for each detection of potential system failure and security breaches, but also applies a powerful framework for continuous monitoring and maintenance of critical water treatment systems.

3.4 Evaluation Metrics

The evaluation metrics used to assess the performance of our model are described below.

3.4.1 Confusion Matrix

A confusion matrix is a tool often used to assess the performance of a classification model. It's particularly useful in machine learning to visualize the accuracy of a predictive model. It's especially useful for assessing models in binary classification tasks, such as determining whether an email is spam or not spam, but it can also be used for multi-class classification problems.Here's a detailed explanation of its components, significance, and utility.

3.4.2 Components of a Confusion Matrix

In a binary classification model, the confusion matrix includes four elements:

- True Positives (TP): Instances correctly predicted as positive.
- True Negatives (TN): Instances correctly predicted as negative.
- False Positives (FP): Instances incorrectly predicted as positive (also known as Type I error).
- False Negatives (FN): Instances incorrectly predicted as negative (also known as Type II error).

These components can be expanded to multiple classes in multi-class classification problems, but the basic principle remains the same—comparing the predicted class against the actual class.

3.4.3 Importance of Confusion Matrices

1. Diagnostic Tool: Helps identify specific areas where the model is making errors, which can guide further improvements in algorithm training.

2. Performance Metrics: Essential values derived from the confusion matrix include:

- Accuracy: Overall, how often is the classifier correct?
- Precision (Positive Predictive Value): When it predicts positive, how often is it correct?
- Recall (Sensitivity or True Positive Rate): How often does it correctly identify actual positives?
- F1 Score: A weighted average of precision and recall, useful when classes are imbalanced.

3. Comprehensive Evaluation: Beyond simple accuracy, it helps evaluate the model in terms of the balance between various error types. This is crucial in applications where some types of errors are more consequential than others.

4 Experiment and Results

4.1 Classification Report

	Precision	Recall	F1-Score	Support
Normal	0.998607	0.998645	0.998626	78945
Attack	0.990304	0.990035	0.99017	11039
Macro Avg	0.994456	0.994340	0.994398	89984
Weighted Avg	0.997588	0.997588	0.997588	89984

The classification report above further reinforces the model's capability to correctly predict both normal and attack classes.

The model achieved impressive results across both classes. For the Normal class, it recorded a precision, recall, and F1-score of 99.86

These metrics reveal that the LSTM model manages the class imbalance effectively, maintaining high precision and recall across both classes. This balanced performance is critical in practical settings where both types of errors—false positives and false negatives—carry significant consequences. The model's overall accuracy stood at 99.76%, with macro-average and weighted-average F1-scores of 99.44% and 99.76%, respectively, reinforcing its efficacy in handling varied operational scenarios within the SWaT dataset.

4.2 Confusion Matrix

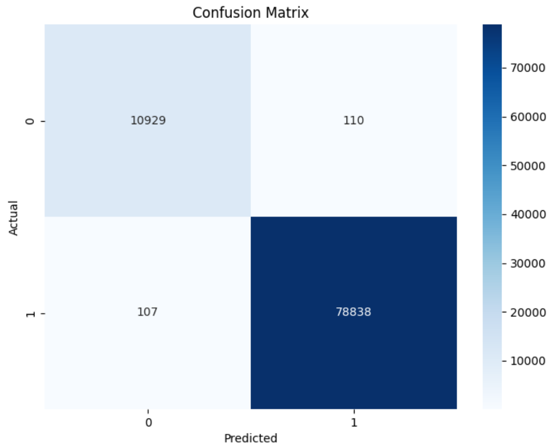
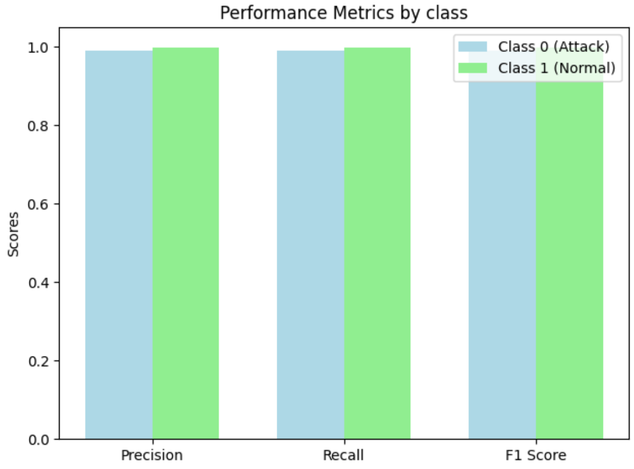


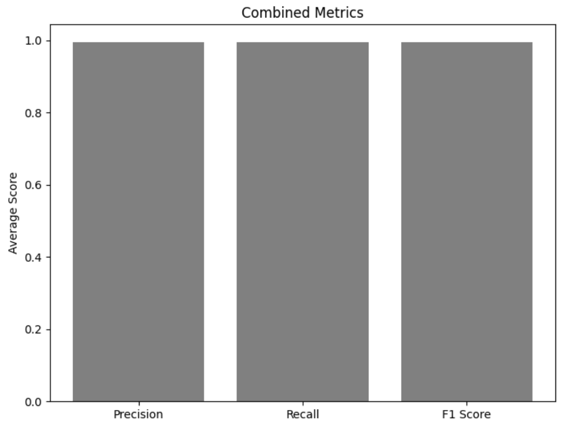
Figure 1: Confusion matrix for our LSTM based approach.

LSTM accurately identified 99.86% of normal class instances and correctly labeled 78,838 true positive cases out of 78,945 in test sample. The high true positive rate along with a very small false negative rate (107 out of 78,945) highlights the model's sensitivity towards identifying the normal operations of the SWaT water treatment facility. This accuracy is essential for the smooth functioning of safety-critical systems. Misclassifying normal system activity as attack or malfunctions results in unnecessary alerts and disruptions to the critical infrastructure.

Similarly, the model's performance in detecting attacks on the physical system was equally impressive. It generated a false positive rate of 0.009% and rarely misclassified an attack class instance as normal. This corresponds to correct labelling of 99.03% of attack observations, highlighting the model's ability to recognize genuine threats. It is vital to maintain system integrity under potential security threats and malfunctions for safety-critical systems.

4.3 Performance Metrics





5 Conclusion

This research has displayed the effectiveness of using Long Short-Term Memory (LSTM) networks for anomaly detection in safety-critical systems, specifically targeting water treatment facilities. The unsupervised learning approach adopted in this study has successfully identified both subtle anomalies and explicit attack attempts; therefore, proving critical for enhancing the security and operational reliability of such systems. Our methodology, centered on a careful preprocessing regimen and the strategic application of LSTM networks, has processed and analyzed data from the Secure Water Treatment (SWaT) testbed, yielding high precision and recall rates. Additionally, the use of confusion matrices and other performance metrics have provided a deeper insight into the model's capabilities, confirming its validity across various scenarios.

Future work will focus on refining these models through the integration of more complex datasets and potentially incorporating real-time data feeds to further enhance predictive capabilities. Additionally, exploring the fusion of LSTM with other machine learning techniques could yield even more robust anomaly detection systems.

Overall, this research project not only advances the field of cybersecurity in water treatment systems but also contributes a significant technological stride towards safeguarding other similar infrastructures against attacks.

6 Github

https://github.com/mariamrafique1/COMP-562-Final-Project-.git

References

[1] Sridhar Adepu, Aditya Mathur, Khurum Nazir Junejo. ”A Dataset to Support Research in the Design of Secure Water Treatment Systems.” October 2016.

[2] Ailin Deng, Bryan Hooi. ”Graph Neural Network-Based Anomaly Detection in Multivariate Time Series.” 2021.

[3] Narhede, Sarang. ”Understanding Confusion Matrix.” 2018.

[4] Laghrissi, F., Douzi, S., Douzi, K. et al. Intrusion detection systems using long short-term memory (LSTM). J Big Data 8, 65 (2021). https://doi.org/10.1186/s40537-021-00448-4. 2020.