This paper presents machine learning approaches to detect cyberattacks on electric grid transformers. Specifically, the authors compare the effectiveness of multiple anomaly detection techniques to detect malicious tampering to transformers’ differential protective relays. The anomaly detection techniques use different types of autoencoder neural networks, where each focuses on a particular type of transformer fault. The authors train and validate their models using the D6 benchmark test system. The results show that their LSTM model outperforms the others on one-phase-to-ground and two-phase-to-ground faults. However, for three-phase-to-ground fault, the other models tend to detect anomalies better. The authors also evaluate their models’ performance on unseen anomalies and find that a Linear AutoEncoder model performs best. The reviewers thought the authors addressed an important topic in security for smart grid systems, and rigorously validated their results using a standard benchmark test system.

Public review written by

David Irwin
University of Massachusetts Amherst, USA
Data Analytics for Cybersecurity Enhancement of Transformer Protection

MARTIYA ZARE JAHROMI, University of Toronto, Canada
AMIR ABIRI JAHROMI, University of Toronto, Canada
DEEPA KUNDUR, University of Toronto, Canada
SCOTT SANNER, University of Toronto, Canada
MARTHE KASSOUF, Hydro-Quebec Research Institute (IREQ), Canada

Electric power substations are experiencing an accelerated pace of digital transformation including the deployment of LAN-based IEC 61850 communication protocols that facilitate accessibility to substation data while also increasing remote access points and exposure to complex cyberattacks. In this environment, machine learning algorithms will play a vital role in cyberattack detection and mitigation and natural questions arise as to the most effective models in the context of smart grid substations. This paper compares the performance of three autoencoder-based anomaly detection systems including linear, fully connected, and convolutional autoencoders, as well as long short-term memory (LSTM) neural network for cybersecurity enhancement of transformer protection. The simulation results indicated that the LSTM model outperforms the other models for detecting cyberattacks targeting asymmetrical fault data. The linear autoencoder, fully connected autoencoder and 1D CNN further outperform the LSTM model for detecting cyberattacks targeting the symmetrical fault data.

CCS Concepts: • Computer systems organization → Embedded and cyber-physical systems; • Computing methodologies → Anomaly detection.

Additional Key Words and Phrases: cybersecurity, data analytics, machine learning, transformer protective relays

1 INTRODUCTION

The rapid integration of standard and interoperable information and communication technologies (ICT) in substations [12, 33] has accelerated the frequency and complexity of electric utility cyberattacks [13]. Attacks against electric power substations such as that on the Ukrainian grid in 2015 have caused significant societal and economic damage including loss of life [10, 28, 30]. As such, the North American Electric Reliability Corporation (NERC) has taken initial steps towards safeguarding cyber-assets by mandating the critical infrastructure protection (CIP) standards [1].

The emergence of standardized and interoperable communication protocols such as IEC 61850 and industrial internet of things (IIoT)-based applications renders traditional security-by-obscurity and perimeter defense security strategies obsolete [2]. Yet, these transformations facilitate accessibility to high fidelity substation data to lay a powerful groundwork for developing machine learning-based data analytics for cybersecurity enhancement [20]. Cybersecurity of substations has been analyzed in the literature from two perspectives; 1) cybersecurity risk assessment/impact analysis [7, 18, 26] and 2) cyberattack detection, mitigation and prevention [5, 15–17, 29]. Most of the approaches solely focus on information technology (IT) data. For instance, some of these approaches attempt to detect cyberattacks by examining the intruders’ footprints on the communication packets. This is while the cyberattack signatures on the operational technology (OT) data have been commonly neglected. This trend is expected to rapidly change in the coming years by the introduction of novel cyberattack detection systems that rely on both information technology (IT) and operational technology (OT).

Anomaly-based techniques have three main advantages over misuse-based techniques. First, anomaly-based techniques can adaptively learn the time varying dynamics and operating points of power systems to establish comprehensive baselines for system behaviors. Second, anomaly-based detection techniques only require training on normal (non-attack) data, which is available in abundance compared to cyberattack data making it possible to easily obtain the necessary training sets for model optimization. Third, anomaly-based techniques are capable of detecting unencountered zero-day cyberattacks. The primary disadvantage of anomaly-based techniques is the potential for high false detection rates because previously unseen system behaviors can be categorized as anomalies [4, 11, 34].

Machine learning-based anomaly detection systems have been extensively examined for cyberattack detection in smart grids. An artificial intelligence-based approach has been proposed in [23] to identify compromised meters. An intrusion detection system has been proposed in [3] for wide area measurements. An unsupervised anomaly detection system has been proposed in [21] to differentiate cyberattacks from disturbances and faults in smart grids. A machine learning-based method has been proposed in [6] to detect cyberattacks against state estimation. In [32], the margin setting algorithm has been employed to defend smart grids against false data injection attacks.

Despite the considerable potential of machine learning-based anomaly detection systems, they have received less attention in the literature compared to analytical approaches for cybersecurity enhancement of substations due to the lack of high fidelity data in traditional substations. A 1-dimensional convolutional based autoencoder has been employed in [24] to identify cyberattacks against distance protective relays. A fully connected autoencoder has been employed in [19] to enhance the cybersecurity of the transformer differential protection. In [4], data analytics comprising long short-term memory neural network and ridge based regression classifier have been used to identify the root causes of the transmission protection mal-operation. Yet, different autoencoder-based anomaly detection systems for cybersecurity enhancement of protective relays have not been compared previously in the literature.
This paper compares the performances of different autoencoder-based anomaly detection systems as well as LSTM for cybersecurity enhancement of transformer protection. Specifically, we employ a variety of autoencoder-based anomaly detection systems as well as the LSTM neural network for cybersecurity enhancement of transformer protection using OT data. The performance of different autoencoder-based anomaly detection systems and the LSTM neural network for identifying different types of cyberattacks are measured and compared.

2 THE FALSE DATA INJECTION ATTACK AGAINST TRANSFORMER PROTECTION

Cyberattackers may target confidentiality, integrity, or availability (C-I-A) of data. Confidentiality aims to prevent users/devices from accessing unauthorized data. Integrity is about validity and correctness of data. Availability deals with the accessibility of data within a reasonable amount of time to an authorized user/device. Availability and integrity of data are paramount for OT systems like protective relays because they rely on real time data to identify abnormal conditions such as faults to actuate circuit breakers. Prominent examples of cyberattacks on the availability and integrity of data include distributed denial of service (DDoS) and false data injection (FDI) attacks, respectively.

The main objective of the proposed anomaly detection systems is to detect the malicious tampering of current measurements by an attacker to illegitimately trigger different elements of the differential protective relay of a transformer. The differential protective relay of a transformer is designed to detect different types of faults including three-phase-to-ground, two-phase-to-ground, and single-phase-to-ground. Therefore, we design and train a separate anomaly detection system for each type of fault. Each of the anomaly detection systems becomes operational by the activation of the corresponding element of the differential protective relay. We further investigate the possibility of considering a universal architecture for anomaly detection systems for different types of faults by comparing the architectures obtained for each type of fault.

Moreover, the merging units receive the commands from the differential protective relay in the form of GOOSE packets through the process bus and send the trip signals to the circuit breakers after performing the digital-to-analog conversion.

An FDI attack is considered here where the intruder manipulates the magnitude and phase angle of the current measurements, yielding different elements of the differential protective relay to issue false tripping commands to the transformer circuit breakers. Multiple scenarios can be considered for the execution of the FDI attack against transformer protection including: 1) the installation of malicious firmware on the merging units through a supply chain attack or by physical access to the merging units, and 2) the injection of false data to the process bus through remote rogue connections using stolen legitimate substation operator credentials.

3 MACHINE LEARNING-BASED ANOMALY DETECTION SYSTEMS

The main objective of the proposed anomaly detection systems is to detect the malicious tampering of current measurements by an attacker to illegitimately trigger different elements of the differential protective relay of a transformer. The differential protective relay of a transformer is designed to detect different types of faults including three-phase-to-ground, two-phase-to-ground, and single-phase-to-ground. Therefore, we design and train a separate anomaly detection system for each type of fault. Each of the anomaly detection systems becomes operational by the activation of the corresponding element of the differential protective relay. We further investigate the possibility of considering a universal architecture for anomaly detection systems for different types of faults by comparing the architectures obtained for each type of fault.

The choice of machine learning models for anomaly detection depends on the nature and dimensionality of the input data. The input data to the anomaly detection system for differential protective relays is composed of two time series of three-phase current measurements. This results in high dimensionality of the input data and complicates feature extraction for machine learning. Moreover, the evolving and clandestine nature of cyberattacks limit the possibility of effective modeling of anomalous behaviour of cyberattacks in contrast to normal behaviour in substations for which there is significantly more data and more predictable characteristics. In this environment, semi-supervised and unsupervised machine learning approaches are in a superior position for cyberattack detection in contrast to supervised machine learning approaches. Advances in semisupervised and unsupervised machine learning has made it possible to solve classification problems, including anomaly detection, with high-dimensional data sets that can suffer from complex structure, sparsity or overfitting [25]. The autoencoder-based anomaly detection systems and LSTM neural network learn to compress the input data into a smaller latent space, then reconstruct the input data from the latent space with a low reconstruction error. Since we train the autoencoders and LSTM neural network with benign or attack-free current measurement sequences, we expect to observe high reconstruction error when feeding malicious current measurement sequences as input [31]. We define the reconstruction error...
for a data sequence $X_t$ as given in (1). A data sequence is considered anomalous if the reconstruction error is above a predefined threshold as given in (2).

$$MSE_t = ||X_t - M(X_t)||^2_2$$

$$MSE_t > \epsilon \rightarrow \text{anomalous data sequence}$$  \hspace{1cm} (1)

where $MSE_t$ is the Mean Squared Error, $X_t$ denotes the input data sequence, $M$ denotes the autoencoder or LSTM model, $M(X_t)$ denotes the output data sequence, and $\epsilon$ denotes the threshold considered on the reconstruction error for anomaly detection.

**Linear Autoencoders.** A linear autoencoder consists of an input layer, a code layer with a size smaller than input/output layers, and an output layer. In a linear autoencoder, all the activation functions in each layer are linear. The linear autoencoder model is similar to dimensionality reduction in Principal Component Analysis (PCA).

**Fully Connected Autoencoders.** In fully connected neural networks, all the neurons in each layer are connected to all the neurons in the subsequent layer. From a technical perspective, a fully connected autoencoder consists of two parts; an encoder and a decoder, as illustrated in Fig. 2. An encoder consists of an input layer, a variable number of hidden layers, and a code (embedding) layer. The code layer connects the encoder and decoder and its size is smaller than input and output layers. The decoder consists of the same number of hidden layers as the encoder and an output layer.

**1D Convolutional Neural Networks.** Unlike fully connected networks, neurons of each layer in CNN are not connected to all the neurons in the following layer and parameter sharing exists that reduces storage. 1D-CNN is a good candidate for anomaly detection because it is capable of detecting localized anomalies due to its window-based nature. Different types of layers are used in CNN autoencoders including convolutional, pooling, and upsampling [25, 27]. The convolution layer works based on convolution operation as given in (3).

$$S(i) = (X * f)(i) = \sum_{j=0}^{n} X(j)f(i - j)$$  \hspace{1cm} (3)

where $X$ denotes the input of the operation, $S$ denotes the output, $f$ denotes the convolution filter and $n$ denotes the length of the convolution filter.

In convolution layers, various filters are applied in parallel to the input to produce a set of linear activations. Each linear activation is followed by a non-linear activation function. To reconstruct the original input in the decoder, upsampling and convolution layers are combined. This combination is also known as transposed convolution or deconvolution. Fig 3 shows an example of an upsampling operation.

**Long Short-Term Memory Networks.** Other types of networks, called recurrent networks, consist of neurons that have self-connections or connections to neurons from previous layers. This recurrence provides the ability for the network to retain what happened in the past (short-term memory). The new state $h_t$ is expressed as:

$$h_t = f_w(h_{t-1}, x_t)$$  \hspace{1cm} (4)

where $x_t$ is the input vector at time step $t$, $h_{t-1}$ denotes the old state, and $f_w$ is a function with parameters $w$.

Consider Fig. 4 as a simple example of an RNN. Using the recurrent formula in (4) at each time step, we can process a sequence of vectors $X = \{x_1, x_2, ..., x_n\}$ using the same function $f$ and weights $w$ at every time step.

To address the vanishing gradient problem in RNNs, Long Short Term Memory (LSTM) networks have been designed. An LSTM network is a good candidate for anomaly detection because it is capable of detecting non-local, long term anomalies. Fig. 5 shows an LSTM unit. The horizontal line on top of the unit is responsible for passing the cell state which facilitates the long-term memory for relevant components of the data. LSTM unit consists of three gates. Forget gate is responsible for removing the parts of the cell state that are no longer needed. Input gate adds the information needed to the cell state. Output gate produces the output. It is possible to stack up arrays of LSTM units to enable more complex LSTM networks.

4 TRAINING, VALIDATION AND OPTIMIZATION OF THE ARCHITECTURE OF THE ANOMALY DETECTION SYSTEMS

In this section, we provide information about the test system and the training data set. We explain the approach employed for optimizing the architectures of the proposed machine learning-based anomaly detection systems. In addition, a set of metrics are presented for measuring the performance of the anomaly detection systems.

4.1 Test System

The IEEE power system relaying committee (PSRC) D6 benchmark test system is considered for generating the training data sets [14]. This test system connects a power plant with four 250 MVA generator units to a 230 kV transmission network through two parallel 500 kV transmission lines. The 230 kV transmission network is modeled as an infinite bus. Differential protective relays protect the power plant transformers as illustrated in Fig. 6.
4.2 Training Data Set

OPAL-RT HYPERSIM is employed to implement and simulate the PSRC D6 test system and generate the training data sets. The simulations are performed for a duration of 1.5 seconds with the fault start varying randomly between t=1 s to t=1.02 s to ensure the fault occurs at different parts of the current waveforms. Note that the period of one cycle is approximately 0.0167 s in a 60 Hz power system. Moreover, the generation levels are changed between 550 MW and 360 MW in 2 MW step size in each simulation to generate data sets under different operating conditions. The simulations are performed for different types of faults including three-phase-to-ground, two-phase-to-ground and single-phase-to-ground faults. The fault impedance is assumed to be zero. In total, 20,736 simulations are performed to generate training data sets for each type of fault. The anomaly detection systems are trained with 80% of the 20,736 simulations for each type of fault. The validation and test data sets each comprises 10% of the 20,736 simulations.

The three-phase current measurements are collected from CT1 and CT2 at the sampling rate of 4800 samples per second in compliance with IEC 61850-9-2 standard for SV packet specifications [9]. An important parameter for training of the machine learning-based anomaly detection system is the input data length. A sliding window of 10 ms, i.e., 48 samples of current measurements per phase, is fed to the anomaly detection systems as input. As such, the input data to the anomaly detection systems contain 6 x 48 = 288 samples in total. In order to obtain the input data, we extracted a 20 ms window from each 1.5 s simulation containing 47 samples before the starting point of the fault and 47 samples after the starting point of the fault. Next, we slide the 10 ms input window of the anomaly detection systems over the 20 ms window of data sample by sample. This amounts to 48 windows of input data per simulation with 10 ms duration.

4.3 Optimizing the Architecture of the Machine Learning-Based Anomaly Detection Systems

We used the grid search method for hyperparameter tuning and optimizing the architecture. In this method, we consider values in Table 1 for each hyperparameter. Different possible combinations of hyperparameters are then tested using grid search. The best hyperparameter values are selected based on the validation error observed in the grid search.

The test data set includes 2074 simulation data sets. We replaced 207 of the test data sets, i.e., 10% of the test data sets, with FDI cyberattack data sets in order to create an imbalanced test data set. We use an imbalanced test data set because cyberattacks in power systems are rare events compared to normal behavior. The hyperparameters considered and tested for each model are listed in Table 1.

The FDI cyberattack data sets considered cover various situations ranging from naive scenarios where the cyberattacker only understands the principles of transformer differential protective relays to very sophisticated cyberattacks where the cyberattacker has some knowledge of power system dynamics and transformer fault signatures.

The cyberattack data are generated by OPAL-RT HYPERSIM. We considered three different scenarios for cyberattack data generation. In the first scenario, random false data are generated by OPAL-RT HYPERSIM with the appropriate magnitude to mimic a fault condition. In the second scenario, the tap setting of the current transformer in the test system are modified in OPAL-RT HYPERSIM.
such that transformer differential protective relay receives current measurements with larger magnitude, mimicking a fault condition. In the third scenario, a fault condition is simulated in OPAL-RT HYPERSIM and used as a replay attack.

### 4.3.1 Linear Autoencoder Architecture
We employed a fully connected autoencoder with one hidden layer, one input layer, and an output layer. All the activation functions in the model are linear.

### 4.3.2 Fully Connected Autoencoder Architecture
We used a fully connected autoencoder with the same number of hidden layers in encoder and decoder. In the encoder part, the number of neurons in hidden layers monotonically decreases from the input to the code layer. We used the Adam optimizer for model optimization. Fig. 7 shows the fully connected autoencoder architecture obtained for anomaly detection. For the sake of brevity, the autoencoder architectures obtained for different types of faults are summarized in Table 2.

### 4.3.3 1D Convolutional Neural Network Architecture
In the 1D CNN architecture, the first layer is zero padding. In the encoder part, we use convolution and max-pooling layers. In the decoder part, there are deconvolution layers, a combination of upsampling and convolution layers. Max pooling and upsampling layers both have window sizes of 2. Filter size for convolution layers is tuned in the hyperparameter tuning step. Table 3 summarizes the details of the architecture obtained for anomaly detection. For conciseness, the hyperparameters selected for 1D CNN are summarized in Table 2.

#### 4.3.4 LSTM Architecture
The many to many one direction LSTM network is considered with input is a sequence of 48 vectors of size 6. We considered stacks of LSTM unit arrays followed by a dense layer. The learning rate, number of LSTM layers, and the size of LSTM unit arrays are tuned in the hyperparameter tuning step and summarized in Table 2. Fig. 8 shows the LSTM architecture obtained for anomaly detection considering the current measurements triggering the two-phase-to-ground fault element of the differential protective relay. $X_t$ represents the current sample at the time step $t$, $Y_t$ represents the output of time step $t$, $h_t$ represents the hidden state, and $c_t$ represents the cell state.

The similarity of architectures obtained for anomaly detection systems for different types of faults in Table 2 indicates that a universal architecture can be possibly designed for different types of faults.

## 5 SIMULATION RESULTS
The performance of the anomaly detection systems is measured using precision and recall metrics, which are more appropriate for imbalanced datasets. It is worth noting that the accuracy metric is not helpful because cyberattacks are rare events. The correct selection of the threshold value plays a vital role on the performance.

### 5.1 Performance Analysis of Autoencoder-Based Anomaly Detection Systems
The performance of the linear autoencoder, fully connected autoencoder, 1D convolutional autoencoder, and LSTM are measured for detecting cyberattacks against different elements of the transformer differential protective relay. We use the precision-recall curve to understand the performances of the four models for different possible thresholds.

### Table 2. Selected Parameters for The Proposed Models. (A: One-Phase-To-Ground Faults, B: Two-Phase-To-Ground Faults, C: Three-Phase-To-Ground Faults)

<table>
<thead>
<tr>
<th>Model</th>
<th>Parameter</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear AE</td>
<td>Learning rate</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>Code Size</td>
<td>40 50 40</td>
<td>40 50 40</td>
<td>40 50 40</td>
</tr>
<tr>
<td>Fully Connected AE</td>
<td>Learning rate</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>Depth</td>
<td>2 3 2</td>
<td>2 3 2</td>
<td>2 3 2</td>
</tr>
<tr>
<td></td>
<td>Code Size</td>
<td>40 40 40</td>
<td>40 40 40</td>
<td>40 40 40</td>
</tr>
<tr>
<td>1D CNN</td>
<td>Learning rate</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>Depth</td>
<td>2 1 2</td>
<td>2 1 2</td>
<td>2 1 2</td>
</tr>
<tr>
<td></td>
<td>Convolution filter size</td>
<td>48 28 48</td>
<td>48 28 48</td>
<td>48 28 48</td>
</tr>
<tr>
<td></td>
<td>Filter count</td>
<td>48 28 48</td>
<td>48 28 48</td>
<td>48 28 48</td>
</tr>
<tr>
<td>LSTM</td>
<td>Learning rate</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>Depth</td>
<td>1 1 1</td>
<td>1 1 1</td>
<td>1 1 1</td>
</tr>
<tr>
<td></td>
<td>LSTM units count</td>
<td>40 40 40</td>
<td>40 40 40</td>
<td>40 40 40</td>
</tr>
</tbody>
</table>

### Table 3. 1D Convolutional Network Architecture for Anomaly Detection in One-Phase-To-Ground Fault Measurements

<table>
<thead>
<tr>
<th>Index</th>
<th>Layer Type</th>
<th>Output Dimensions</th>
<th>Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Input</td>
<td>Size:288</td>
<td>Length:1</td>
</tr>
<tr>
<td>2</td>
<td>Zero Padding</td>
<td>292 1</td>
<td>pad size: 3</td>
</tr>
<tr>
<td>3</td>
<td>Convolution</td>
<td>292 32</td>
<td>filter size: 6</td>
</tr>
<tr>
<td>4</td>
<td>Max pooling</td>
<td>146 32</td>
<td>window size: 2</td>
</tr>
<tr>
<td>5</td>
<td>Convolution</td>
<td>146 64</td>
<td>filter size: 6</td>
</tr>
<tr>
<td>6</td>
<td>Max pooling</td>
<td>73 64</td>
<td>window size: 2</td>
</tr>
<tr>
<td>7</td>
<td>Convolution</td>
<td>73 128</td>
<td>filter size: 6</td>
</tr>
<tr>
<td>8</td>
<td>Up sampling</td>
<td>146 128</td>
<td>window size: 2</td>
</tr>
<tr>
<td>9</td>
<td>Convolution</td>
<td>146 64</td>
<td>filter size: 6</td>
</tr>
<tr>
<td>10</td>
<td>Up sampling</td>
<td>292 64</td>
<td>window size: 2</td>
</tr>
<tr>
<td>11</td>
<td>Convolution</td>
<td>292 32</td>
<td>filter size: 6</td>
</tr>
<tr>
<td>12</td>
<td>Up sampling</td>
<td>292 1</td>
<td>window size: 2</td>
</tr>
<tr>
<td>13</td>
<td>Cropping</td>
<td>288 1</td>
<td>size: 288</td>
</tr>
</tbody>
</table>

### Table 4. Performance of the Anomaly Detection Systems. (A: One-Phase-To-Ground Faults, B: Two-Phase-To-Ground Faults, C: Three-Phase-To-Ground Faults)

<table>
<thead>
<tr>
<th>Model</th>
<th>Prec.</th>
<th>Rec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear AE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fully connected AE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1D CNN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LSTM</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Fig. 7. Autoencoder structure for three-phase-to-ground fault.*
The LSTM model outperforms the other three models for the one-phase-to-ground and two-phase-to-ground faults as illustrated in figures 9 and 10. However, the linear autoencoder, fully connected autoencoder and 1D CNN approximately have similar curves and outperform the LSTM model for the three-phase-to-ground fault as illustrated in Fig. 11. Table 4 summarizes the results when the threshold is selected such that the precision is equal to 1. It is worth noting that even a subtle change in the performance of anomaly detection systems for protective relays is significant because the misoperation due to cyberattacks has the potential to cause major disturbances and widespread blackouts in power systems. Given the symmetry of three-phase-to-ground faults and asymmetry of single-phase-to-ground and two-phase-to-ground faults, we conclude that LSTM performs better for asymmetrical faults and is weaker than the other models for symmetrical faults. We feel that such a trend will generalize to other more complex systems beyond the benchmark system employed in this paper because of the inherent ability of LSTM to recognize time series patterns and manage long-term memory patterns in contrast to the other models.

5.2 Impact of Data Granularity on Anomaly Detection Performance

In this study, we investigate the impact of generation level granularity on the performance of each of the machine learning algorithms while considering the three-phase-to-ground-fault. Thus, test data sets are generated for the three-phase-to-ground-fault with finer generation level granularity compared to the training data set, i.e.; the generation levels are changed with 1 MW step size in each simulation. Next, we measured the performance of the linear autoencoder, fully connected autoencoder, 1D convolutional autoencoder and LSTM for detecting cyberattacks while considering cyberattack data and data of finer generation level granularity that have not been considered in the training step.

Table 5 summarizes the results when the threshold is selected such that the precision is equal to 1. The comparison between the results in Table 4 and Table 5 show that the performance of all four models significantly drops when they are exposed to data captured from other generation levels that are not included in the original dataset. Yet, the linear autoencoder model outperforms the three other models.

6 CONCLUSION

This paper presented four machine learning-based anomaly detection systems including linear autoencoder, fully connected autoencoder, convolutional autoencoder, and LSTM neural network for cybersecurity enhancement of transformer differential protection for anomaly detection in transformer relays. The simulation results underscore that the LSTM model outperforms the other models for one-phase-to-ground and two-phase-to-ground faults. The linear autoencoder, fully connected autoencoder and 1D CNN further outperform the LSTM model for the three-phase-to-ground fault. The impact of input data granularity on the performance of the deep learning-based anomaly detection systems is further investigated using a sensitivity analysis. The results showed that the performance of the all four models significantly drop when they are exposed to the previously unseen system behaviors. Yet, the linear autoencoder model outperformed the three other models when it is exposed to the previously unseen system behaviors.

REFERENCES


