Anomaly detection in load forecasting using ARIMA and Autoencoder

Arun Abhishek I, Dafinny T

2023 IEEE Fifth International Conference on Advances in Electronics, Computers and Communications (ICAECC) | 979-8-3503-0544-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICAECC59324.2023.10560105

***Abstract*—Machine learning (ML) has seen widespread acceptance as a result of recent advancements in a variety of power system applications, including meter data analytics; renewable, load, and price forecasting; and grid security evaluation. The operation and planning of electricity systems depends on load forecasting. By using input features such as historical load consumption data, we can build load forecasting models to guide decision-making in unit commitment and economic dispatch. AutoRegressive Integrated Moving Average (ARIMA) models have been widely used in industry for time series forecasting. In this paper, we present load forecasting models using the ARIMA and autoencoder models. Then, we investigate the vulnerability and safety of the above algorithms in the power system. The assumption is that the attacker does not need to be familiar with the underlying power system or the load forecasting methodology. Using four cyber attack templates, we study the effect of perturbations that require specific assumptions on the ability of the adversaries. Then, the 3-sigma rule is tested, which is a widely used heuristic for anomaly detection. Additionally, we investigate the detection of anomalies introduced by cyber attacks using Autoencoders.**

***Index Terms*—Anomaly detection, ARIMA model, Autoencoder, Load forecasting, Machine learning**

I. Introduction

Machine learning utilizes data and algorithms to make predictions without being explicitly programmed to do so. In order to make predictions, machine learning algorithms build a model using sample data, also referred to as training data. The algorithm works better when there is more data available. Many different fields, including medicine, email filtering, speech recognition, and image classification, use machine learning. Data, such as numbers, images, or text, is collected and prepared to be used as training data for machine learning. The machine learning algorithm then learns from the data to enhance its performance on a particular task after programmers select a model to use and provide the training data.

Unsupervised learning is a method of machine learning in which models are trained using unlabeled datasets. Contrary to supervised learning, where models are taught to map the input to the output based on labelled data, unsupervised learning algorithms learn patterns from unlabeled data. Unsupervised machine learning algorithms goal is to actively discover patterns that can be inferred from the data. Applications for unsupervised machine learning can be used to complete tasks like dimensionality reduction, anomaly detection, and clustering. However, compared to supervised learning, unsupervised learning faces some particular difficulties, such as the inability to assess the algorithm's performance.

Machine learning (ML) can be used for load forecasting tasks. ML-based forecasting methods can be divided into two classes: regression based ML and classification based ML. ML can be a useful tool for forecasting tasks, especially when dealing with large and complex datasets. ML models can learn patterns and relationships in the data that can be used to make accurate predictions. However, the performance of the ML model depends on the quality and quantity of the training data, as well as the choice of the appropriate ML algorithm and hyperparameters. The economic and reliable performance of electrical systems can both benefit from the precision and authenticity of load forecasting data. To make operational choices and plans under diverse power grid situations, power system operators significantly rely on the information on load forecasts. Supervisory Control And Data Acquisition (SCADA) used at control center can be subjected to cyber attacks. As an illustration, malicious entities opened substation breakers and attacked the Ukraine power grid in 2015 [1]. This has garnered significant attention into the security of power systems by cyber attacks. With the advancement of technology, utilities, and system operators must place greater importance on accuracy and robust forecast.

Time series forecasting is a useful tool for load forecasting, as it can help electrical power system planners and operators to make informed decisions about future load demands. The choice of the appropriate time series forecasting method depends on the specific requirements of the load forecasting task, such as the forecast horizon and the accuracy required. Time series forecasting, in statistical terms, is the process of analyzing time series data using statistics and modeling to make predictions and inform strategic decisions. To make close to accurate forecasts, we need to acquire the time series data over a period, analyze the data, and then build a model that will help us make the forecast. In this study, for the sake of simplicity, we limit the time series inputs of the model to historical load. Support vector regression [2], ARIMA [3], and neural networks [4] are only a few statistical and machine learning methods that have been used successfully in practice for short-term load forecasting. An ML model such as the ARIMA model has been found to have a straightforward

Arun Abhishek I, Dafinny T are with the Department of Electrical and Electronics Engineering, Sri Venkateswara College of Engineering (SVCE), Post Bag no 1, Pennalur, Sriperumbudur, Tamil Nadu, 602117, India. Email:(arun.abhishek1995@gmail.com, dafinny1922@gmail.com)

application in load forecasting.

Time series anomaly is defined first. Anomaly detection for time series involves finding outlier data points compared to some standard or typical signal values. There are three categories of anomalies from the perspective of time series data, namely point, contextual, and collective anomalies. These anomalies can introduce unanticipated spikes, dips, and trend changes. As forecasting techniques are becoming more advanced, the vulnerabilities of model to cyber security risks is less explored. In this paper, we look into the vulnerability of the ML model to cyber attacks.

Time series forecasting models considered here are Autoencoder and ARIMA models, and is subjected to four different types of cyber attack templates, namely pulse attack, scale attack, ramping attack, and random attack, which are discussed in this paper [5]. Additionally, we evaluate the potential impact of the proposed attack and construct an attack detection strategy to find time series anomalies or outliers by determining whether the actual load data lies within the thresholds of forecasting models.

In this paper, we propose both regression and classification model, wherein ARIMA and autoencoder can predict the future load demand and the target class of the data sample. The classification model predicts the probability that each instance belongs to one class or another. The performance measures in machine learning classification models are used to assess how well machine learning classification models perform in a given context. These performance metrics include accuracy, precision, recall, and F1-score. It helps us understand the strengths and limitations of these models.

The rest of the paper is organized as follows. In Section II we review the relevant literature related to the study; in Section III the templates of considered cyberattacks is discussed; in Section IV load forecasting models such as ARIMA and autoencoder is described, In section V we discuss on the results of the experiments; finally in Section VI, we draw conclusions with discussions on the security and robustness of the ML model in power systems.

II. Literature review

In this section, we furnish a brief literature review on load forecasting, the robustness of current ML algorithms, and power system cyber-security. Our work is different from most related work in two aspects: most of the studies on load forecasting address the improvement of accuracy of the model [3]; we analyzed the cyber-security of power systems considering four attack templates [5], and then we extended a simple but effective ARIMA or autoencoder approach for detection.

This research study is connected to the substantial corpus of research on forecasting in power networks, including forecasting for loads [6] and power output from renewable energy sources. A few percentage points of forecast error reduction can lead to better system operation. In load forecasting challenges, a variety of approaches have been used and assessed, including nonparametric regression, support vector regression [2], ARIMA, and neural networks. Among these autoregressive integrated moving average (ARIMA) [7] and neural network [4] models have straightforward applications in load forecasting. Both ARIMA and autoencoder models can be used for short-term forecasts and can handle non-stationary time series data. In order to increase the robustness of the ML algorithm, our study focuses on developing an ML algorithm coupled with the anomaly detection technique.

ARIMA models with more intricate feature representations is encouraged in electricity price forecast models [7]. The approach used to forecast next-day power prices is based on the ARIMA methodology. In order to analyze time series data and create forecasting models for short-term electricity prices, the article uses ARIMA techniques. The effectiveness of the ARIMA models is assessed by the authors, who also compare them to other forecasting techniques. The article comes to the conclusion that the ARIMA models can enhance decision-making in the power market by offering precise and trustworthy forecasts of next-day electricity prices. The study adds to the body of knowledge on predicting power costs and shows how ARIMA models can be used to predict electricity prices over the short term. In this research, we focus on comprehending the robustness of ARIMA model by combining it with anomaly detection.

Machine learning-based anomaly detection (MLAD) methodology is proposed for load forecasting under cyberattacks [1]. It highlights the importance of detecting cyberattacks on power systems. The authors develop an MLAD model that uses machine learning techniques to detect anomalies in load forecasting data, which can indicate the presence of a cyberattack. The paper evaluates the performance of the MLAD model and compares it with other anomaly detection methods. The cyber attacks for load forecasting were divided into five categories: pulse, scaling, ramping, random, and smooth curve. These categories serve as attack templates in this paper except smooth curve attack.

Descriptive analytics-based anomaly detection method is posited for cyber secure load forecasting [8]. This study's main goal is to create descriptive analytics-based methods (DABM) for anomaly detection in order to safeguard the load forecasting process from cyberattacks especially long sequence anomalies. They suggest an Integrated Solution (IS) and a Hybrid Implementation of IS (HIIS) that can identify and reduce long sequence anomalies brought on by cyberattacks. When compared to IS, HIIS is also capable of significantly lowering false positive rates and increasing true positive rates. The proposed HIIS can be used as a system for forecasting load. The suggested approach makes use of descriptive analytics to spot data anomalies and pinpoint their possible causes.

Adaptive robust regression is another suggested technique for identifying and reducing the impact of cyberattacks on load forecasting data [9] . It emphasizes on the need for techniques that can identify and lessen the impact of cyber attacks on load forecasting data. In order to perform load forecasting in the linear regression setting, it was suggested that other statistical method such as robust linear regression

method be used instead. The benefit of using such techniques is that the defender need not be aware of the attackers' attack template. Three approaches were taken into consideration: ALTS, M-estimation, and L1 regression. When the proportion of attacked data is high, the comparison study using the GEF-Com2012 dataset indicates that the ALTS method significantly outperforms M-estimation methods in terms of forecasting accuracy, and the robustness does not degrade as the proportion of attacked data rises. The proposed method is shown to be effective in the experimental findings for identifying and reducing the impact of cyberattacks on load forecasting data.

The importance of high-quality real-time load data for achieving accurate load forecasting in the smart grid is highlighted in [10]. This paper proposes a model-based anomaly detection method that consists of two components, a dynamic regression model and an adaptive anomaly threshold. Then, a real-time anomaly detection method based on the DRM for the corrupted load data is detailed, which can be further cleansed by replacing the detected anomalies with the forecasted hourly load from the last sliding simulation. Finally, a general framework is proposed for future research on anomaly detection for load forecasting. The paper presents experimental results that demonstrate the effectiveness of the proposed method in detecting anomalies in load data used for very short-term load forecasting.

III. CYBERATTACKS ON LOAD FORECASTS

A brief description of the five categories of cyberattacks used for load forecasting is described namely pulse, scaling, ramping, random, and smooth curve in [5]. The attacker's objective is to increase or decrease anticipated values. The attacker can use the above 5 templates to skew the output forecast values in one direction.

1) Pulse Attack: In pulse attack, load projections are changed to higher/lower levels at a certain time. The attack parameter is set as λ_p.

$$\dot{p_F^t} = (1 + \lambda) * p_f^t for t = t_p \quad (1)$$

where t_p is the occurrence time of one pulse attack.p_f^t is the original load forecast that is not tampered with any cyber attack.$\dot{p_F^t}$ is the load forecast tampered with cyber attacks.

2)Scaling Attack: Scaling attacks alter the values over a set period of time after multiplying them by a scaling attack parameter λ_s.

$$\dot{p_F^t} = (1 + \lambda) * p_f^t for t_s < t < t_e \quad (2)$$

where t_s and t_e stand for a cyberattack's beginning and ending times, respectively.

3) Ramping Attack:

Ramping attacks come in two distinct varieties. Type I ramping attacks only take up-ramping anomalies into account. The values are multiplied by a ramping function with the specified range $\lambda_R t$.

$\dot{p_F^t}$=λ_R*(t- t_s)*p_f^t for t_s <t <t_e

Both up- and down-ramping anomalies are taken into account in Type II ramping attacks. The equation shown below is slightly different from [5] as it was observed that this particular equation introduces up and down ramp anomalies.

$\dot{p_F^t}$=[1+λ_R*(t_s -t)]*p_f^t for t_s <t <[(t_s +t_e) /2]

$\dot{p_F^t}$=[1+λ_R*(t_e- t)]*p_f^t for [(t_s +t_e) /2]<t <t_s where t_s and t_e stand for a cyberattack's beginning and ending times, respectively.

4) Random Attack:

In this approach, forecasts are loaded using positive numbers added from a uniform random function.

$\dot{p_F^t}$=p_f^t +λ_{RA}*Rand(t) for t_s <t <t_e

where rand is a uniformly distributed random number generator.is a scale factor and defined as half of the maximum of load forecast value, i.e., λ_{RA}= max(p_f^t)/2. It is assumed that attackers set the commencement and finish times of each random attack at random.

5) Smooth-Curve Attack: The original forecasted data's contiguous start and end points are replaced to implement smooth-curve attacks. In this study, the original forecasted data is swapped out for nearby points and a smooth curve is produced using polynomial fitting.

IV. MODEL FORECASTING

For forecasting time series, two separate techniques are used: ARIMA and autoencoder. A statistical technique called ARIMA (Autoregressive Integrated Moving Average) makes predictions about the future of a time series based on its historical values. It combines the weighted sum of previous values with moving averages and differencing to create AR (autoregressive) predictions. The autoencoder, on the other hand, uses deep learning to learn the latent representations of input data as random variables.

A. ARIMA(Autoregressive Integrated Moving Average)

The ARIMA, model effectively develops a linear equation that explains and predicts your time series data. Three different components that make up this equation are AR, I, and MA.

Based on historical values ARIMA models forecast future values. Lagged moving averages are used by ARIMA to smooth time series data. The equation for ARIMA is

Y_t=α+β1Y_{t-1}+β2Y_{t-2}+.....+$\beta p Y_{t-p}\epsilon_t$+$\phi$1$\epsilon_{t-1}$+$\phi$2$\epsilon_{t-2}$+.. +$\phi q \epsilon t - q$

where Y_{t-1} is the lag1 of the series, β is the coefficient of lags that the model estimates, and α is the intercept term, also estimated by the model.

Finally, the ARIMA model is almost always represented as ARIMA(p, d, q). The p and q is determined from autocorrelation function (ACF) and partial autocorrelation function (PACF). p is the order of the autoregressive (AR) component, d is the order of differencing, and q is the order of the moving average (MA) component.

There are a few steps that have to be followed in order to determine p, d, and q: [11]

- check for the stationarity
  There are a number of statistical tests that may be used to check for stationarity in ARIMA equations. One such test is Augmented Dickey-Fuller (ADF) test is another

assessment that determines whether the series has a unit root, which explains whether the series is stationary.
If the time series is stationary try to fit the ARMA model, and if the time series is non-stationary then seek the value of d. The Box-Jenkins method, conditional least squares, and maximum likelihood estimation are a few techniques for fitting an ARMA model. One common method is the Box-Jenkins method. The Box-Jenkins method involves several steps, including plotting the data, examining the autocorrelation function (ACF) and partial autocorrelation function (PACF) to determine the appropriate values of p and q for the ARMA(p,q) model.
If a time series is non-stationary, then the value of d can be determined by taking the difference of the time series d times until it becomes stationary. The order of differencing, d, is an important parameter in the ARIMA model. If the data is getting stationary then draw the autocorrelation and partial autocorrelation graph of the data as shown in Fig. 1.

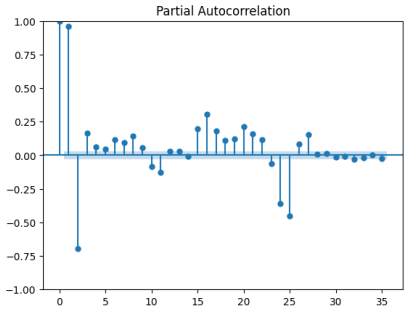

Fig. 1. Sample Partial Autocorrelation plot

B. AUTOENCODER

Autoencoder models can be implemented for anomaly detection especially when the input data has a large dimensionality, complex structure, or both.

1) AUTOENCODER: An autoencoder is an ANN that consists of two independent components: the encoder, which changes the original data using the formula h = f(x), and the decoder, which reconstructs the data using the formula x = g(h) as shown in Fig. 2.

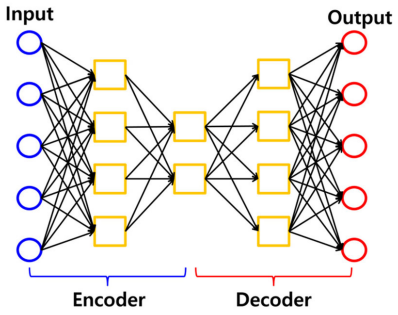

Fig. 2. Architecture of Autoencoder

A few procedures must be followed for accurate data forecasting [12]: Designing the autoencoder architecture, Training the autoencoder, Encoding the data, Forecast.

V. METRICS FOR FORECASTING AND CLASSIFICATION

A. Accuracy of forecasting models

The commonly used accuracy metrics to measure performance of forecasting are [13]: Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE).

In this, MAPE (Mean Absolute Percentage Error) is a commonly used method in ARIMA forecasting because of the following advantages: 1) it is a relative error metric, which means that it is not affected by the scale of the data. 2) It is easy to interpret, as it represents the average percentage difference between the predicted and actual values.

The formula for the MAPE is [14]:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}|A_t - F_t/A_t| \tag{3}$$

In the formula above:

$\sum$ indicates to sum of values, n is the sample size, A_t is the actual value at that time instance, F_t is the predicted value at that time instance.

MAPE indicates model prediction performance ; The lower is the value of MAPE, the better model is performing.

B. Anomaly classification

To forecast the target class of the data sample in classification problems, classification models are used. The likelihood that each instance belongs to a particular class is predicted by the classification model. To effectively use classification models in production for resolving practical issues, it is critical to assess their performance. In this paper for the purpose of classification in autoencoder accuracy, precision, recall, and F1-score are some of these performance metrics that are being used.

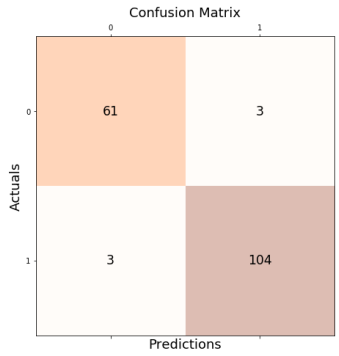

Fig. 3. Confusion matrix

In Fig. 3, the predicted data results could be interpreted as follows [15]:

- True Positive (TP): The degree to which the model accurately predicts the positive class is measured by true positive.In this case, True Positive has a value of 104.
- False Positive (FP): When the model predicts that an instance belongs to a class when it doesn't, this is known as a false positive. Consequently, in this case, False Positive has a value of 3.
- True Negative (TN): The outcomes that the model correctly identifies as negatives are known as true negatives.In this case, True Negative has a value of 61.
- False Negative (FN): When a model predicts something as negative when it is actually positive, it is called a false negative. Consequently, False Negative has a value of 3.

1) Precision Score: The percentage of labels that were correctly predicted positively is represented by the model precision score. A model with high precision is the one we would pick if we wanted to reduce false negatives. The precision score is a useful measure of the success of prediction when the classes are very imbalanced.

Precision Score = TP / (FP + TP)

2) Recall Score: The model's ability to correctly predict positives out of real positives is measured by the model recall score. This differs from precision, which counts the proportion of accurate positive predictions among all positive predictions made by model.

Recall Score = TP / (FN + TP)

3) Accuracy score: The ratio of true positives and true negatives to all positive and negative observations is referred to as the model accuracy, which is a performance metric for machine learning classification models.

Accuracy Score = (TP + TN)/ (TP + FN + TN + FP)

4) F1-Score: The model score as a function of recall and precision is represented by the model F1 score. A substitute for accuracy metrics, the F-score is a machine learning model performance metric that equally weights precision and recall when assessing how accurate the model is.

F1 Score = 2* Precision Score * Recall Score/ (Precision Score + Recall Score/)

VI. RESULT

The data for this particular study consists of actual load consumption data from ISO New England website for a period of six months(Jan 1,2022 to June 30, 2022). [16] and programs utlized for this study are developed in python. This particular programming language was developed by Guido van Rossum in 1980's [17].The training data for ARIMA and Autoencoder models are Jan 1,2022 to June 20,2022. The testing data consist June 21,2022 to June 30,2022.

ARIMA model order used (2,1,1), which is determined using auto_arima function of pmdarima in python. The Auto encoder's layers and units are chosen arbitrarily. The encoder consists of 3 layers with 64,32 and 16 with relu activation units. The decoder consists of 3 layers with 16,32 and 64 units respectively. Min max scaler is used to scale the input data to a range of (0,1). The 'adam' optimizer and 'mse'(mean squared error) is chosen as metric for the same

The plot of forecasted values of aggregated power transmission level demand for ARIMA and Autoencoder is shown in Fig. 4 and Fig. 5 respectively. The MAPE values for ARIMA and Autoencoder are 12.2% and 1.5%. Auto encoder has better forecasting accuracy, which can be inferred from the plot as well.

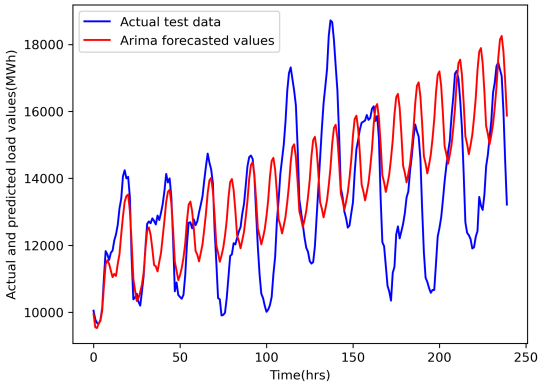

Fig. 4. ARIMA forecasted values

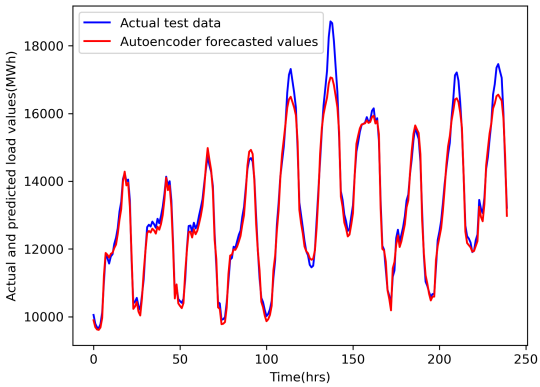

Fig. 5. Autoencoder forecasted values

Gaussian 3-sigma rule is used as threshold to determine anomalies for both the forecasting models, wherein the mean and standard deviation of train dataset is computed. The lower, upper threshold is computed by the subtraction and addition of thrice the standard deviation to mean. It is experimentally determined for this training dataset that the thresholds are 19206.81, 6024.56 MWh (MegaWatthour) respectively.

The forecasted values for both models are subjected to different attack templates. The value of Lambda is chosen as 0.1. The type 2 ramping consist of upscaling and downscaling attacks. The confusion matrix for the same is shown in Figures below

Fig. 6 and Fig. 7 shows the confusion matrix for pulse attack and ramp1 attack respectively. It is observed that there's a false negative of 1 in both matrices, which denotes that there's one anomalous point, which was not detected correctly. Additionally, a true positive of 5 in Fig. 8 denotes that model has correctly identified the five anomalous points

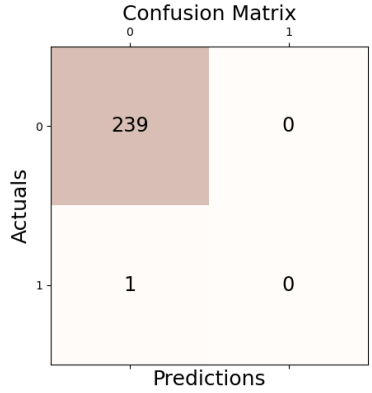

Fig. 6. Confusion matrix for pulse attack on Auto encoder forecasts

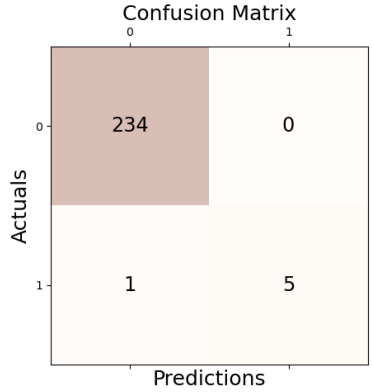

Fig. 7. Confusion matrix for ramp1 attack on Auto encoder forecasts

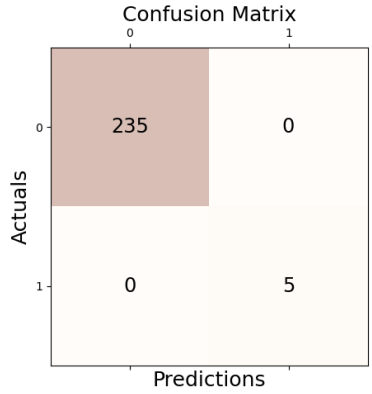

Fig. 8. Confusion matrix for random attack on Auto encoder forecasts

Table I shows different performance metrics of prediction of the anomaly using an ARIMA and Autoencoder. The metrics are obtained are same for both the models. Both models accurately determines ramp1 and random attacks.

VII. CONCLUSION

Load forecasting using ARIMA and Autoencoder was studied for a real world ISO New England dataset. Four different cyber attack templates are considered, which introduces anomalies in the forecasted load values. A gaussian 3 sigma based approach is used for anomaly detection. It is observed the performance metrics are same for anomaly detection for both models. But Auto encoder models have a better forecasting accuracy. Future research can probe into different thresholds for anomaly detection.

Attack	Accuracy	precision	recall	F1-score
scale	0.975	0.0	0.0	0.0
pulse	0.996	0.0	0.0	0.0
ramp1	0.996	1.0	0.833	0.909
ramp2	0.979	0.0	0.0	0.0
random	1.0	1.0	1.0	1.0

TABLE I
PERFORMANCE METRICS OF ALGORITHMS TO DIFFERENT ATTACKS

REFERENCES

[1] "The 2015 ukraine blackout: Implications for false data injection attacks," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 3317–3318, 2017.

[2] E. Ceperic, V. Ceperic, and A. Baric, "A strategy for short-term load forecasting by support vector regression machines," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4356–4364, 2013.

[3] Y. Chen, Y. Tan, and B. Zhang, "Exploiting vulnerabilities of load forecasting through adversarial attacks," in *Proceedings of the tenth ACM international conference on future energy systems*, 2019, pp. 1–11.

[4] K. Chen, K. Chen, Q. Wang, Z. He, J. Hu, and J. He, "Short-term load forecasting with deep residual networks," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 3943–3952, 2019.

[5] M. Cui, J. Wang, and M. Yue, "Machine learning-based anomaly detection for load forecasting under cyberattacks," *IEEE Transactions on Smart Grid*, vol. 10, no. 5, pp. 5724–5734, 2019.

[6] D. Park, M. El-Sharkawi, R. Marks, L. Atlas, and M. Damborg, "Electric load forecasting using an artificial neural network," *IEEE Transactions on Power Systems*, vol. 6, no. 2, pp. 442–449, 1991.

[7] J. Contreras, R. Espinola, F. Nogales, and A. Conejo, "Arima models to predict next-day electricity prices," *IEEE Transactions on Power Systems*, vol. 18, no. 3, pp. 1014–1020, 2003.

[8] M. Yue, T. Hong, and J. Wang, "Descriptive analytics based anomaly detection for cybersecure load forecasting," *IEEE Transactions on Smart Grid*, vol. PP, pp. 1–1, 01 2019.

[9] Z. Tang, J. Jiao, P. Zhang, M. Yue, C. Chen, and J. Yan, "Enabling cyberattack-resilient load forecasting through adversarial machine learning," in *2019 IEEE Power & Energy Society General Meeting (PESGM)*, 2019, pp. 1–5.

[10] J. Luo, T. Hong, and M. Yue, "Real-time anomaly detection for very short-term load forecasting," *Journal of Modern Power Systems and Clean Energy*, vol. 6, 01 2018.

[11] "Time series forecasting using arima." [Online]. Available: https://towardsdatascience.com/time-series-analysis-arima-based-models-541de9c7b4db

[12] "Autoencoder." [Online]. Available: https://towardsdatascience.com/lstm-autoencoder-for-anomaly-detection-e1f4f2ee7ccf

[13] "Arima models." [Online]. Available: https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/

[14] "Mape." [Online]. Available: https://datagy.io/mape-python/

[15] "Performance metrics." [Online]. Available: https://vitalflux.com/accuracy-precision-recall-f1-score-python-example/

[16] "Iso new england load data." [Online]. Available: https://www.iso-ne.com/isoexpress/web/reports/load-and-demand/-/tree/dmnd-rt-hourly-sys

[17] "Python programming language." [Online]. Available: https://en.wikipedia.org/wiki/Python$_{(programming_language)}$