2022 IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS) | 979-8-3503-9645-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/IoTaIS56727.2022.9975951

Online Location-based Detection of False Data Injection Attacks in Smart Grid Using Deep Learning

Hanem I. Hegazy*[1], Adly S. Tag Eldien*[2], Mohsen M. Tantawy‡[3], Mostafa M. Fouda§¶[4], and Heba A. TagElDien*[5].
*Department of Electrical Engineering, Faculty of Engineering at Shoubra, Benha University, Cairo, Egypt.
‡Network Planning Department, National Telecommunication Institute (NTI), Cairo, Egypt.
§Department of Electrical and Computer Engineering, Idaho State University, Pocatello, ID, USA.
¶Center for Advanced Energy Studies (CAES), Idaho Falls, ID, USA.
Emails: [1]Hanem.Hegazy@feng.bu.edu.eg, [2]adlytag@feng.bu.edu.eg, [3]Ntimohsen@gmail.com, [4]mfouda@ieee.org, [5]HEBAALLAH.SHAHAT@feng.bu.edu.eg.

***Abstract*—The smart grid is a multi-dimensional data-generating cyber-physical system. Distributed architectures and the heterogeneous nature of the Internet-of-Things (IoT) sensors make it more prone to various cyber-attacks. False data injection attacks (FDIAs) have recently emerged as significant threats to smart grid state estimation. As a result, real-time locational detection of stealthy FDIAs is critical for smart grid security and reliability. In this paper, we introduce a comparative analysis of various deep-learning approaches to test their effectiveness in the location-based detection of FDIA. Also, a deep learning approach is developed by constructing a multi-feature architecture based on a convolution neural network and long short-term memory network (MCNN-LSTM). Extensive testing on IEEE test cases has demonstrated that the proposed approach outperforms the existing deep learning approaches in locating FDIAs for small and large systems under different attack scenarios. We evaluate the performance of each model in terms of presence and location-based detection accuracy, model complexity, and prediction time. Extensive results in the IEEE 14 and IEEE 118-bus systems show that the suggested architecture has a locational detection accuracy of more than 94% and 95%, respectively. From the results, we can conclude the proposed approach is more robust, scalable, and faster in detecting the locations of compromised measurements than the other deep learning models.**

***Index Terms*—Smart Grid, FDIA, CNN, LSTM, Bi-LSTM, CNN-LSTM**

I. Introduction

A smart grid (SG) is a cyber-physical system (CPS) that combines modern digital technologies, automation, computer, and control to enable bidirectional communications between consumers and utility [1]–[7]. Despite the several advantages of having more control and data over the power network, SGs are vulnerable to cyber-attacks [8]–[14]. The primary objective of cyber-attacks is to disrupt or maliciously mislead the SG state estimation process, which could result in regional blackouts or attempts to manipulate the price of electricity with catastrophic economic consequences [15]–[19]. One of these cyber-attacks is data integrity attacks. FDIAs are considered severe types of data integrity attacks that a power network can be vulnerable to. FDIAs attempt to deceive power system status estimations by introducing fake data into meter measurements [20], [21]. FDIAs might be well-structured (stealthy) or unstructured attacks. Several researchers have proposed different approaches to construct FDIA [22]. A well-structured FDIA can be performed even if the attacker only may have some partial knowledge of the power grid configuration information [23].

Two strategies can be used to protect the network from such attacks: protection-based strategy and detection-based strategy. Recently, detection-based approaches are based on deep learning (DL) models. Deep neural networks have demonstrated their ability to detect anomalies in supervised [24], semi-supervised [25], and unsupervised [26] domains. In [27], the authors developed an enhanced DBN architecture called Conditional Deep Belief Network (CDBN) to extract high-dimensional temporal aspects of FDIAs. The authors in [28] adopted this concept and added a Convolutional Neural Network (CNN) architecture in front of the Recurrent Neural Network (RNN) to adapt the data's dimensionality. An LSTM-based encoder-decoder for anomaly identification with an F1-score of above 0.84 is presented in [29]. In [30], the authors proposed a multilabel based classification strategy for location detection of FDIA using CNN. The research in [31] introduces a CNN with LSTM, a CNN with the Gated Recurrent Unit (GRU), and K-Nearest Neighbors (KNN) strategies for the detection of FDIA. The main propositions of this research can be summarized as follows:

- a comparative evaluation of several deep learning models has been conducted to identify the best multilabel classifier for locating compromised meters under a variety of attack and topology scenarios.
- a multi-feature based detector based on one-dimensional CNN (Conv1D) and LSTM is proposed. All of the research was done using the IEEE 14-bus and 118-bus test systems, which yielded acceptable findings for presence and location detection accuracy.

Extensive evaluations have been performed in terms of the presence and detection-based accuracy, model complexity, and prediction time.

The remainder of this paper is structured as follows: Section II presents the power system model for state estimation. The problem formulation and suggested architectural designs are described in Section III. The performance of the suggested detectors is then demonstrated in section IV. The paper's conclusion is included in Section V.

II. PROBLEM FORMULATION

A. State estimation Model

Power systems are constantly checked in order to maintain normal and secure operating conditions. This is accomplished by employing the state estimation function. We assumed that there are M meters with M measurements $z_1, ..., z_M$ and N state variables $x_1, ..., x_N$. The relationship between these M meter measurements and N state variables. A linear regression model for state estimation using the DC power flow model can express the relationship between these M metre measurements and N state variables as follows [32]:

$$z = \boldsymbol{H}x + e, \tag{1}$$

where e represents measurement errors with a zero mean. Whereas $\boldsymbol{H} \in R^{mxn}$ denotes Jacobian matrix. The associated measurement will be considered poor data as long as

$$R = \|z - \boldsymbol{H}\hat{x}\|_2 \geq \tau \tag{2}$$

B. Stealthy False Data Injection Attack (FDIA)

In the case of stealthy FDIA, Ref. [23] proved that an optimally structured attack can be formed by solving a min-cut problem when the attacker only has minimal knowledge of the measuring matrix $\boldsymbol{H}$. Any non-zero arbitrary vector can be chosen by the attacker as the attack vector a, and then construct the malicious data added to the initial measurements a= $\boldsymbol{H}c$, where $c \neq 0$ and $c \in R^n$ be any arbitrary vector [33]. The measurements vector can then be represented as:

$$z_a = \boldsymbol{H}x + e + a \tag{3}$$

Such attacks can deceive detection by conventional residual test techniques in Eq. (2), As a result, the control unit may assume that the compromised state $\hat{x_a} = (\hat{x} + c)$ is the actual state and in such circumstances, the residual remains unaffected:

$$\|z_a - \mathbf{H}\hat{x_a}\| = \|z + a - \boldsymbol{H}(\hat{x} + c)\| = \|z - \boldsymbol{H}\hat{x}\| \tag{4}$$

III. FDIA LOCATION-BASED DETECTION SCHEMES

This section provides the detection mechanisms and how they will be implemented.

A. Detection of FDIA location

Identifying the location of the FDIA vector is accomplished by categorizing all the meters' measurements into two labels: 1 for the compromised meter and 0 for an uncompromised meter. Therefore, the locational detection of FDIA can be characterized as a multilabel classification problem.

B. Multi-feature base CNN-LSTM Architecture (MCNN-LSTM)

The developed methodology can handle measurement datasets with meter malfunctions, failures, communication issues, unstructured and structured FDIA. The suggested architecture is shown in Fig.1 and composed of:

1) Permute Block

When permutation is applied to the input before the CCN block, just a single step with N variables will be processed. Because the input now consists of multi-features readings, The data can be constructed using a tensor of shape (B, N, M), where B denotes the batch size, N is the number of steps, and M is the number of measurements processed for each time step.

2) CNN Block

Assuming that L CNN layers, on each of these layers, a set of 1D filters x is applied. The first convolutional layer's feature maps $m_{1,j}$ which were created using the multi-feature input measurements z, can be expressed as

$$m^t{}_{1,j} = \mathrm{ReLU}\left(z^t * x_{1,j} + b_{1,j}\right) \tag{5}$$

where $x_{1,j}$ is the j$^{\text{th}}$ kernel, and $b_{1,\text{j}}$ is the corresponding bias. $b_{1,\text{j}}$ is added to all the convolution output, and the convolution process is indicated by $*$. These feature maps are then fed into the LSTM layer. The inputs for I$^{\text{th}}$ convolutional layer are the features generated at $(\text{I}-1)^{\text{th}}$ LSTM Layer. The output of the I$^{\text{th}}$ layer is as follows:

$$m^t_{I,j} = \mathrm{ReLU}\left(m^t_{I-1,j} * x_{I,j} + b_{I,j}\right) \tag{6}$$

where $m^t_{I,j}$ represents the jth feature map at the I-th convolutional layer.

3) LSTM Block

An LSTM is a modified RNN structure. According to [34], the vanishing gradient problem that plagues conventional RNNs is avoided in LSTM by incorporating gating functions into its state dynamics. LSTM has multi-feature maps from CNN layer m^t at each time step. Three gates: an input gate i, a forget gate f, and an output gate o are presented in each LSTM cell. The information flow of an LSTM cell is as follows:

$$f_t = \sigma_g\left(w_f m_t + u_f h_{t-1} + b_f\right) \tag{7}$$

$$i_t = \sigma_g\left(w_i m_t + u_i h_{t-1} + i\right) \tag{8}$$

$$o_t = \sigma_g\left(w_o m_t + u_o h_{t-1} + b_o\right) \tag{9}$$

$$\tilde{\mathrm{c}}_t = \tanh\left(w_c m_t + u_c h_{t-1} + b_c\right) \tag{10}$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{\mathrm{c}}_t \tag{11}$$

$$h_t = o_t \cdot \tanh\left(c_t\right) \tag{12}$$

where w is the recurrent connection between the previous and current hidden layers. The weight matrix that connects the inputs to the hidden layer is denoted by u. $\tanh$ denotes tangent function, and . represents element-wise multiplication. Here, *h* and *c* represent hidden state and cell state vectors, respectively. The LSTM layer's features are then passed to another CNN layer. The I$^{\text{th}}$ output of the LSTM layer is then passed to a global pooling layer.

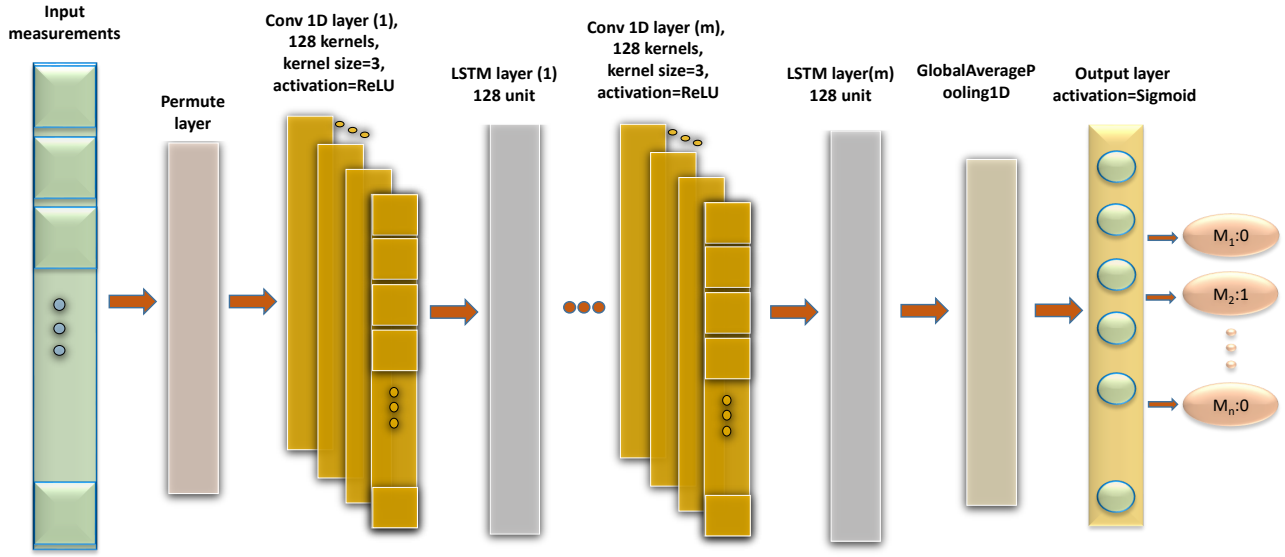

Fig. 1. FDIA's MCNN-LSTM location-based detector architecture.

4) Fully connected layer

Finally, a dense layer is linked to the n outputs, and the output layer is employed with a sigmoid function to classify the meter labels. The final multi-label categorization output for every meter j at t time step can be shown as:

$$\hat{y}^t = \sigma_g \left(w_d \times h_t + b_d \right) \tag{13}$$

where, σ_g denotes the sigmoid function, and w_d, b_d are the weights and biases of the fully connected layer, respectively.

C. Training Procedure

The training data is composed of the measurement vector z and the meter labels y^t. These labels are used to train the detectors and can be shown as $y^t = 1$ for compromised meters or $y^t = 0$ for otherwise.

The key parameters, including the number of kernels, neurons, activation function, optimizer, learning rate, batch size, number of epochs, and number of layers, must be modified before the recommended detector can be used to classify the measurements from the meters. The number of layers utilized may have an impact on the accuracy of locational identification; using too few layers may result in underfitting, while using too many layers may result in overfitting. Finding the appropriate parameters for the suggested approach during the training phase is the aim of the parameter-tuning procedure.

1) Mini-batch, early stopping, and cross-validation technique

Cross-validation, the early stopping technique, and mini-batch gradient descent are employed to reduce over-fitting and speed up the convergence of the detector architectures. The gradient descent is calculated for each mini-batch using 100 randomly chosen examples from the training dataset. For each batch, the training dataset is separated into three portions: 0.7 for training, 0.3 for cross-validation which means .2 used for validation, and 0.1 for testing.

2) Loss function

Cross entropy is primarily employed for multilabel classification, which shows the error between real meter labels and predicted meter labels for each mini-batch. During training, the loss function is used to optimize the hyper-parameters and is represented as follows:

$$dL_{i=} \sum_{t \in \theta} \frac{-1}{m} \sum_{i=1}^{m} \hat{y}_i \log y_i^t + (1 - \hat{y}_i) \log \left(1 - y_i^t\right) \tag{14}$$

IV. EXPERIMENTAL RESULTS

This section specifically focuses on the training and testing datasets. Following that, evaluation metrics for FDIA detection were mentioned. Also, this section investigates the effectiveness and reliability of the suggested scheme's performance in identifying the attack presence and falsified meters' location.

A. Dataset

This section evaluates the proposed locational detector from FDIA in IEEE 14- and 118-bus systems. The grid topologies are available from MATPOWER package and The power topologies can be described as follows: for the IEEE 14-test case, the number of total measurements is 19 while in the IEEE 118-test case, the number of total measurements is 180.

1) Model's Parameters

The Keras library and Tensorflow are used to train the proposed model (MCNN-LSTM) which utilizes 128 filters with 3x1 kernel sizes, and a 'RELU' activation function for each Conv1D layer. 128 hidden unit is employed for each layer of LSTM. Furthermore, the epoch is set at 200. The batch size is set to 100 and the cross entropy is employed as the loss function for prediction. The data is fitted using the Adam optimizer, which has starting learning rate of 0.001 and patience of 5. We compared the proposed approach to state-of-the-art models such as multilayer perceptron

(MLP), CNN, LSTM, Bidirectional LSTM (Bi-LSTM), and CNN-LSTM.

2) Training, and Testing Dataset

The dataset for input measurements is adopted from [30] and prepared as follows:

- The training set has a dimension of 550,000 × Bus_Size. The training data are divided into 500, 000 samples without attack and 50,000 samples under attack.
- Five different L2-Norms of the injected attacks (attack levels range from 1 to 5) are included in the testing set, each with a dimension of 10,000 Bus Size for measurements and labels. Each variety is divided into 5,000 uncompromised samples and 5,000 compromised samples.

The classifier's outputs $\hat{y}^t$ are continuous numbers ranging from 0 to 1. As a result, the classifier establishes a distinction threshold to decide whether to classify the output as 0 or 1. The discrimination threshold can be changed to increase or decrease the sensitivity to application parameters. The discrimination threshold is set at 0.5.

We assessed the offered approaches when the L2-Norm FDIA is 1 and the standard deviation of the measurement error is 0.2. The number of hidden layers varies between one and four. We used the same datasets for training and testing to ensure a fair comparison.

B. Evaluation Metrics

The Precision, Recall, and F1-score of the predicted labels were employed as performance indicators. The precision and recall are represented as follows

$$\text{Precision} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Positive (FP)}}, \tag{15}$$

$$\text{Recall} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Negative (FN)}}, \tag{16}$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{17}$$

Another important evaluation criterion for locating compromised meters is row/location accuracy (RACC). RACC is described as the likelihood that the detector would classify all of the meters locations that are without attack as uncompromised, but meters under attack are labeled as compromised [30].

1) IEEE 14-test case

The metrics for the deep learning models with varying numbers of hidden layers are compared in table I for the IEEE 14-test case. Overall, the MLP model has the worst locational detection accuracy and least F1 Score. F1-score of CNN, LSTM, Bi-LSTM, and CNN-LSTM are above 99.3% and for RACC, CNN is above 93%, LSTM and Bi-LSTM are above 94%, and CNN-LSTM is above 95%. The suggested detector, however, outperforms deep learning models as it achieves F1 Score and RACC values of above 99.5% and 96%, respectively.

We implemented these structures with three hidden layers to achieve an acceptable balance of computational difficulty and locational detection accuracy.

2) IEEE 118-test case

The performance assessment in the IEEE 118-bus test case is provided in table II. Precisions and recalls are always above 99%, which means All models can completely detect the presence of FDIA. It can be observed that MLP has the worst RACC, 61.62%, and 63.57% at layer 1 and layer 4, respectively. Meanwhile, BI-LSTM outperforms CNN, LSTM, and CNN-LSTM. MCNN-LSTM, BI-LSTM, LSTM, CNN-LSTM and CNN reach RACC above 93%, 85%, above 84%, above 84% and 82%, respectively. As a result, MCNN-LSTM performs better than the alternative methods. This shows that even with a complex bus system, the proposed detector can detect both the presence and location of FDIA, proving the scalability of the proposed approach.

3) Robustness

We test the suggested architecture's resilience in the data-collecting environment against the attacker's aggression by lowering the standard deviation of noise to 0.2 and varying the L2 norm of the FDIA from 1 to 5. From Fig. 2, We observe that at variant 1 of L2-Norm, the RACC of CNN, LSTM, Bi-LSTM, CNN-LSTM are 93.31%, 95.05%, 94.95%, and 95.28%, respectively. At variant 5 of L2-Norm, the RACC are 92.06%, 99.74%, 99.88%, 99.81% and 99.87%, respectively. Meanwhile, the MCNN-LSTM detector achieves 96.27% and 99.97%. Overall, the suggested approach outperforms the previous models and is sensitive to low and high L2-Norm of FDIA.

4) Scalability

We examined the scalability of the suggested architecture in the IEEE 118-test case because the performance gap in this system is more noticeable than it is in the IEEE 14-test case. As depicted in Fig.3, Compared to other detection techniques, the suggested MCNN-LSTM detection scheme is more sensitive to lower levels of L2-norm. At variant 1 and variant 5 of L2-norm, MCNN-LSTM archives 92.14% and 99.13% while Bi-LSTM, LSTM, CNN-LSTM, CNN and MLP reach 85.75%, 84.58%, 84.57%, 82.23% and 66.34% at variant 1 and 98.96%, 98.74%, 98.75%, 96.88% and 91.85% at variant 5, respectively. Overall, the locational performance of the MCNN-LSTM detection scheme

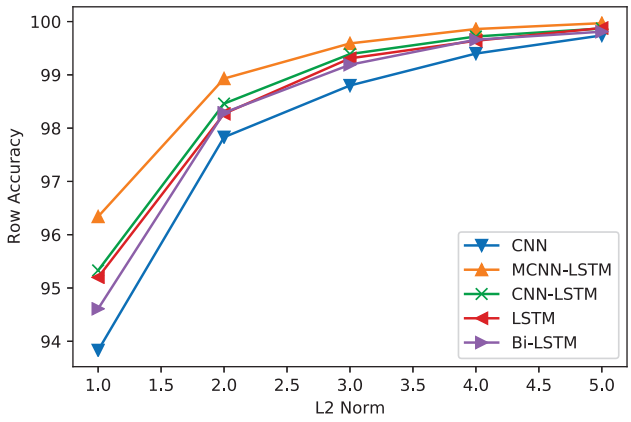

Fig. 2. RACC comparison in the IEEE 14-test case.

TABLE I
PERFORMANCE EVALUATION OF THE IEEE 14-TEST CASE.

Model	Layers	Precision %	Recall %	F1-score %	RACC %	Number of Parameters	Test Time (Sec)
MLP	1	97.83	97.24	97.54	66.16	46,483	0.18
	2	98.10	98.01	98.06	73.97	62,995	0.26
	3	98.26	98.08	98.17	74.85	79,507	0.29
	4	98.20	98.04	98.12	74.57	96,019	0.34
CNN	1	99.21	99.43	99.32	93.28	119,059	0.20
	2	99.19	99.43	99.31	93.54	168,339	0.29
	3	99.16	99.43	99.30	93.83	217,619	0.30
	4	99.10	99.40	99.25	93.52	266,899	0.38
LSTM	1	99.23	99.49	99.36	94.67	113,299	0.19
	2	99.26	99.49	99.38	94.64	245,395	0.30
	3	99.19	99.54	99.37	94.94	377,491	0.37
	4	99.30	99.57	99.44	95.50	509,587	0.45
Bi-LSTM	1	99.20	99.51	99.36	94.56	226,579	0.27
	2	99.20	99.50	99.35	94.60	621,843	0.47
	3	99.21	99.52	99.36	94.66	1,017,107	0.68
	4	99.19	99.53	99.36	94.81	1,412,371	0.90
CNN-LSTM	1	99.28	99.47	99.37	94.51	173,971	0.22
	2	99.27	99.58	99.42	95.36	350,483	0.30
	3	99.27	99.59	99.43	95.44	526,995	0.39
	4	99.27	99.57	99.42	95.43	703,507	0.47
MCCN-LSTM	1	99.46	99.64	99.55	96.12	141,971	0.18
	2	99.42	99.66	99.54	96.30	323,347	0.24
	3	99.43	99.64	99.53	96.31	504,723	0.28
	4	99.45	99.62	99.53	96.07	686,099	0.36

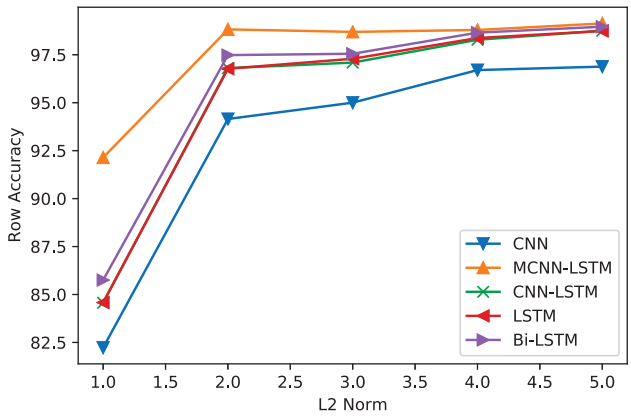

Fig. 3. RACC comparison in the IEEE 118-test case.

outperforms the competitive methods. All the benchmark models can effectively identify the presence of the FDIA as their F1-score is above 99%. We can infer from Table II and Fig. 3 that MCNN-LSTM is more suitable for the 118-bus system than other models.

5) Location Detection

As shown in Table III, the suggested architecture outperforms all models in identifying the locations of compromised meters. We use an IEEE 118-test case to show the location-based detection accuracy for eight attack scenarios as follows:

1) 5th meter is under attack and 6th is not;
2) 6th meter is under attack and 5th is not;
3) 5th and 6th meters Both are under attack ;
4) Neither 5th nor 6th meters are under attack ;
5) 5th meter is under attack and 20th is not;
6) 20th meter is under attack and 5th is not;
7) 5th and 20th meters Both are under attack;
8) Neither 5th nor 20th meters are under attack.

When the system becomes larger with a low L2-norm of FDIA, we can observe that other deep learning models, such as MLP, CNN, LSTM, Bi-LSTM, and CNN-LSTM, are no longer effective to distinguish the co-occurrence dependencies of adjacent meters. In the meanwhile, under a low level of attack, the proposed MCNN-LSTM can locate compromised measurements as the system size grows.

6) Model complexity

There are several benefits to fewer trainable parameters. The first benefit is that the smaller value of the gradient item facilitates quicker training. The second benefit is that dropout layers aren't necessary because overfitting occurs less frequently, as seen in Table I and Table II. Thus, utilizing a smaller set of trainable parameters might reduce model complexity and accelerate model implementation.

For the 118-test case represented in Table II, MLP, CNN, CNN-LSTM, LSTM, and Bi-LSTM models require total parameters of 4,180,660, 4,236,340, 4,504,500, 4,478,644, 4,478,644 and 9,219,252 respectively, to achieve RACCs of 66.11%, 82.07%, 84.16%, 84.72%, and 85.72%. Meanwhile, the suggested approach requires 587,316 parameters to achieve a detection accuracy of 93.56%. When compared to benchmark models, MCNN-LSTM has lower complexity

TABLE II
PERFORMANCE EVALUATION OF THE IEEE 118-TEST CASE.

Model	Layers	Precision %	Recall %	F1-score %	RACC %	Number of Parameters	Test Time (Min)
MLP	1	99.66	99.69	99.67	61.62	4,147,636	0.23
	2	99.64	99.77	99.70	67.73	4,164,148	0.29
	3	99.65	99.74	99.69	66.11	4,180,660	0.34
	4	99.65	99.70	99.68	63.57	4,197,172	0.40
CNN	1	99.79	99.89	99.84	82.33	4,137,780	0.29
	2	99.78	99.89	99.84	82.79	4,187,060	0.40
	3	99.76	99.89	99.83	82.07	4,236,340	0.43
	4	99.79	99.88	99.83	81.83	4,285,620	0.46
LSTM	1	99.84	99.88	99.86	83.84	4,214,452	0.65
	2	99.84	99.89	99.87	84.58	4,346,548	1.79
	3	99.85	99.89	99.87	84.72	4,478,644	1.85
	4	99.85	99.88	99.86	84.78	4,610,740	2.57
Bi-LSTM	1	99.85	99.90	99.86	85.04	8,428,724	1.30
	2	99.84	99.90	99.87	85.66	8,823,988	2.56
	3	99.85	99.90	99.886	85.72	9,219,252	4.16
	4	99.85	99.91	99.88	86.69	9,614,516	5.74
CNN-LSTM	1	99.84	99.89	99.87	84.51	4,233,908	0.70
	2	99.85	99.89	99.87	84.85	4,369,204	1.27
	3	99.85	99.88	99.86	84.16	4,504,500	1.95
	4	99.84	99.88	99.86	84.23	4,639,796	2.65
MCCN-LSTM	1	99.94	99.96	99.95	93.45	224,564	0.23
	2	99.94	99.97	99.95	94.16	405,940	0.33
	3	99.93	99.96	99.95	93.56	587,316	0.39
	4	99.83	99.91	99.87	86.00	768,692	0.48

TABLE III
LOCATION DETECTION (RACC %) ON 5TH, 6TH, AND 20TH METERS.

Compromised Location	MLP	CNN	LSTM	Bi-LSTM	CNN-LSTM	MCNN-LSTM
5th not 6th	69.24	81.74	84.93	86.03	83.95	94.61
6th not 5th	64.53	79.56	83.84	84.49	82.65	93.82
5th & 6th	56.91	74.66	77.23	79.40	77.37	93.50
Neither 5th nor 6th	67.13	83.54	85.73	86.68	85.30	93.38
5th not 20th	63.27	78.29	81.34	82.78	81.09	94.06
20th not 5th	63.57	80.31	85.04	86.34	84.41	93.87
5th & 20th	63.73	78.67	81.07	83.20	80	94.13
Neither 5th nor 20th	67.60	83.54	85.46	86.21	84.89	93.33

and faster implementation.

7) prediction Time

From Table I, it's observed that the MCNN-LSTM model predicts compromised meters more quickly than the other models. Additionally, from Table II it can be observed that MCNN-LSTM has a prediction time that is slightly larger than MLP but MLP is more complex and has a lower detection accuracy. Overall we can conclude Bi-LSTM is the slowest model in predicting the location of compromised meters, whereas MCNN-LSTM detects compromised meters more quickly than CNN, LSTM, Bi-LSTM, and CNN-LSTM models.

V. CONCLUSION

The use of deep neural networks to handle data integrity issues was covered in this research, with a focus on the location-based detection of FDIAs in smart grids for affected measurements. We evaluated various deep learning techniques in a comparative analysis to see how well they could locate compromised measurements. In addition, we proposed a multi-feature based on the CNN-LSTM approach (MCNN-LSTM). The MCNN-LSTM receives inputs as a multi-feature time series with a single time step by the CNN and LSTM blocks. This is accomplished through the dimension permutation layer. The proposed model's resilience, scalability, complexity, and prediction time of the proposed model have been examined through in-depth simulations in IEEE-test cases using TensorFlow and Keras libraries. In particular, the results demonstrated that the MCNN-LSTM can identify the locations of FDIAs throughout the bus systems in a variety of attack scenarios.

REFERENCES

[1] Z. M. Fadlullah, M. M. Fouda, N. Kato, A. Takeuchi, N. Iwasaki, and Y. Nozaki, "Toward intelligent machine-to-machine communications in smart grid," *IEEE Communications Magazine*, vol. 49, no. 4, pp. 60–65, 2011.

[2] M. M. Fouda, Z. M. Fadlullah, N. Kato, R. Lu, and X. S. Shen, "A lightweight message authentication scheme for smart grid communications," *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 675–685, 2011.

[3] M. M. Fouda, Z. M. Fadlullah, N. Kato, A. Takeuchi, and Y. Nozaki, "A novel demand control policy for improving quality of power usage in smart grid," in *2012 IEEE Global Communications Conference (GLOBECOM)*, 2012, pp. 5154–5159.

[4] S. Aladdin, S. El-Tantawy, M. M. Fouda, and A. S. Tag Eldien, "MARLA-SG: Multi-agent reinforcement learning algorithm for efficient demand response in smart grid," *IEEE Access*, vol. 8, pp. 210 626–210 639, 2020.

[5] M. M. Fouda, Z. M. Fadlullah, N. Kato, R. Lu, and X. Shen, "Towards a light-weight message authentication mechanism tailored for smart grid communications," in *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2011, pp. 1018–1023.

[6] M. I. Baza, M. M. Fouda, A. S. Tag Eldien, and H. A. Mansour, "An efficient distributed approach for key management in microgrids," in *2015 11th International Computer Engineering Conference (ICENCO)*, 2015, pp. 19–24.

[7] Z. M. Fadlullah and N. Kato, *Evolution of smart grids*. Springer, 2015.

[8] M. S. Abdalzaher, M. M. Fouda, and M. I. Ibrahem, "Data privacy preservation and security in smart metering systems," *Energies*, vol. 15, no. 19, p. 7419, 2022.

[9] Z. M. Fadlullah, M. M. Fouda, N. Kato, X. Shen, and Y. Nozaki, "An early warning system against malicious activities for smart grid communications," *IEEE Network*, vol. 25, no. 5, pp. 50–55, 2011.

[10] M. M. Fouda, Z. M. Fadlullah, and N. Kato, "Assessing attack threat against ZigBee-based home area network for smart grid communications," in *The 2010 International Conference on Computer Engineering & Systems*, 2010, pp. 245–250.

[11] S. A. Yadav, S. R. Kumar, S. Sharma, and A. Singh, "A review of possibilities and solutions of cyber attacks in smart grids," in *2016 International Conference on Innovation and Challenges in Cyber Security (ICICCS-INBUSH)*, 2016, pp. 60–63.

[12] G. Liang, J. Zhao, F. Luo, S. R. Weller, and Z. Y. Dong, "A review of false data injection attacks against modern power systems," *IEEE Transactions on Smart Grid*, vol. 8, no. 4, pp. 1630–1638, 2017.

[13] M. Faheem, S. Shah, R. Butt, B. Raza, M. Anwar, M. Ashraf, M. Ngadi, and V. Gungor, "Smart grid communication and information technologies in the perspective of industry 4.0: Opportunities and challenges," *Computer Science Review*, vol. 30, pp. 1–30, 2018.

[14] M. I. Ibrahem, M. M. Badr, M. Mahmoud, M. M. Fouda, and W. Alasmary, "Countering presence privacy attack in efficient AMI networks using interactive deep-learning," in *2021 International Symposium on Networks, Computers and Communications (ISNCC)*, 2021.

[15] M. I. Ibrahem, M. Nabil, M. M. Fouda, M. M. E. A. Mahmoud, W. Alasmary, and F. Alsolami, "Efficient privacy-preserving electricity theft detection with dynamic billing and load monitoring for ami networks," *IEEE Internet of Things Journal*, vol. 8, no. 2, pp. 1243–1258, 2021.

[16] M. M. Badr, M. I. Ibrahem, M. Mahmoud, M. M. Fouda, F. Alsolami, and W. Alasmary, "Detection of false-reading attacks in smart grid net-metering system," *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 1386–1401, 2022.

[17] M. I. Ibrahem, M. M. Badr, M. M. Fouda, M. Mahmoud, W. Alasmary, and Z. M. Fadlullah, "PMBFE: Efficient and privacy-preserving monitoring and billing using functional encryption for AMI networks," in *2020 International Symposium on Networks, Computers and Communications (ISNCC)*, 2020.

[18] J. Cloherty and P. Thomas, "Trojan horse," *Bug Lurking in Vital US Computers Since*, 2011.

[19] M. I. Ibrahem, M. Mahmoud, M. M. Fouda, F. Alsolami, W. Alasmary, and X. Shen, "Privacy preserving and efficient data collection scheme for ami networks using deep learning," *IEEE Internet of Things Journal*, vol. 8, no. 23, pp. 17 131–17 146, 2021.

[20] H. I. Hegazy, A. S. Tag Eldien, M. M. Tantawy, M. M. Fouda, and H. A. TagElDien, "Real-time locational detection of stealthy false data injection attack in smart grid: Using multivariate-based multi-label classification approach," *Energies*, vol. 15, no. 14, 2022.

[21] K. Gai, M. Qiu, Z. Ming, H. Zhao, and L. Qiu, "Spoofing-jamming attack strategy using optimal power distributions in wireless smart grid networks," *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2431–2439, 2017.

[22] R. Deng, G. Xiao, R. Lu, H. Liang, and A. V. Vasilakos, "False data injection on state estimation in power systems—attacks, impacts, and defense: A survey," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 411–423, 2016.

[23] S. Bi and Y. J. Zhang, "Using covert topological information for defense against malicious attacks on DC state estimation," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 7, pp. 1471–1485, 2014.

[24] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," *arXiv preprint arXiv:1901.03407*, 2019.

[25] D. Wulsin, J. Blanco, R. Mani, and B. Litt, "Semi-supervised anomaly detection for EEG waveforms using deep belief nets," in *2010 Ninth International Conference on Machine Learning and Applications*, 2010, pp. 436–441.

[26] A. Tuor, S. Kaplan, B. Hutchinson, N. Nichols, and S. Robinson, "Deep learning for unsupervised insider threat detection in structured cybersecurity data streams," *arXiv preprint arXiv:1710.00811*, 2017.

[27] Y. He, G. J. Mendis, and J. Wei, "Real-time detection of false data injection attacks in smart grid: A deep learning-based intelligent mechanism," *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2505–2516, 2017.

[28] Q. Deng and J. Sun, "False data injection attack detection in a power grid using RNN," in *IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society*, 2018, pp. 5983–5988.

[29] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "LSTM-based encoder-decoder for multi-sensor anomaly detection," *arXiv preprint arXiv:1607.00148*, 2016.

[30] S. Wang, S. Bi, and Y.-J. A. Zhang, "Locational detection of the false data injection attack in a smart grid: A multilabel classification approach," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8218–8227, 2020.

[31] D. Mukherjee, S. Chakraborty, and S. Ghosh, "Deep learning-based multilabel classification for locational detection of false data injection attack in smart grids," *Electrical Engineering*, vol. 104, no. 1, pp. 259–282, 2022.

[32] A. Abur and A. G. Exposito, *Power system state estimation: theory and implementation*. CRC press, 2004.

[33] M. Esmalifalak, L. Liu, N. Nguyen, R. Zheng, and Z. Han, "Detecting stealthy false data injection using machine learning in smart grid," *IEEE Systems Journal*, vol. 11, no. 3, pp. 1644–1652, 2017.

[34] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.