LSTM-based Quick Event Detection in Power Systems

Boyu Wang, Yan Li, Jing Yang
School of Electrical Engineering and Computer Science
The Pennsylvania State University
University Park, PA 16802
{*bxw91,yql5925,yangjing*}*@psu.edu*

***Abstract*—In this paper, a data-driven online approach is established to detect events in power systems in real time. The approach does not require prior knowledge of the power system model or its parameters. Instead, it utilizes a long short-term memory (LSTM) model to capture the state evolution of the power system. Due to the expressiveness of the LSTM model, it is able to track the system states with small prediction error when it operates under normal conditions. However, when the system is perturbed by certain events that cannot be predicted by the model, the prediction error will increase dramatically. Thus, by tracking the prediction error of the trained LSTM model, the data-driven online approach is able to detect events in a timely fashion. The event detection problem is then cast into the quick change detection framework, where a Cumulative Sum (CUSUM) based approach is proposed. To overcome the difficulty that the statistics of the prediction error when events happen is generally unknown beforehand, a generalized likelihood ratio test (GLRT) is incorporated into the CUSUM procedure. A Rao-test is then adopted to reduce the computationally complexity of GLRT. Finally, the LSTM based event detection approach is validated with real-world PMU measurements.**

***Index Terms*—Event detection, LSTM, CUSUM.**

I. Introduction

The ubiquitous uncertainties existing in power systems, such as line outage, short-circuit fault, load shedding, generation change, make the stable and reliable operation very challenging. Efficiently detecting those events is crucial for a power system to maintain its stable operation and prevent it from large scale blackouts or cascading failures [1].

On the other hand, the rapid deployment of advanced sensing devices in power systems, such as phasor measurements units (PMUs), makes real-time grid monitoring and control possible. Various online event detection methods have been proposed in recent years [2]–[7]. In order to reduce the detection delay, the problem is often cast into the quickest change detection (QCD) framework [8]–[11]. In [8], QCD is used to detect and identify transmission line outage in near real-time. References [9], [10] adapt the QCD framework to detect false data injection attacks in power grids. In [11], an S-transform based cumulative sum detector is proposed to detect unintentional islanding. In [12], QCD is adopted to monitor the power quality in smart grids. These methods highly rely on either accurate distributional information of the measurements, such as the phase angle change in [8], or knowledge about the power system, such as the grid topology in [9], [10]. However, in practice, time series collected from power grids usually contain complex patterns that cannot be accurately modeled by independent and identically distributed (i.i.d.) random variables. Furthermore, prior knowledge of power system usually is not given. Thus, applying QCD methods directly on raw measurements may lead to degraded detection performance.

Recently, researchers have proposed to resolve this issue by combining time series prediction and event detection [6], [13], [14]. In [6], a state space model is first applied to predict future frequency measurements, and the probabilistic prediction errors are summarized to quantify the uncertainty. Specifically, a probability threshold and a duration threshold are combined to detect the events using the prediction error. In [13], a long short-term memory (LSTM) network is used to predict the time sequence and the prediction error is modeled as a multivariate Gaussian random vector, which is then used to evaluate the likelihood of events. In [14], the normal behavior is first learned by a Kalman filter model, and then the difference between the observation and the predicted value from the model is used to detect events. While such prediction methods are able to capture the temporal dynamics of the power system, the corresponding event detection methods are usually heuristic and may not detect events timely.

In this work, we aim to incorporate time series prediction to the QCD framework. We first adopt an LSTM model to capture the temporal evolution of the measurements when the system operates under a normal condition, which can be trained using historical measurements. We then utilize the trained model to predict upcoming measurements and track the prediction error in real time. We assume that the prediction error before an event follows a normal distribution, whose parameters can be learned based on training data. When an event happens and the system state is perturbed, the trained LSTM model cannot track the state change closely. Thus, the corresponding distribution of the prediction error will change dramatically. Such distribution shift enables us to cast the event detection problem into the QCD framework, which is then

This work was supported in part by the US National Science Foundation under Grant ECCS-1650299 and the College of Engineering of The Pennsylvania State University under a Multidisciplinary Research Seed Grant.

solved efficiently through a CUSUM based method.

There are three main advantages of the proposed LSTM based event detection approach: First, it is purely data-driven and does not require any prior knowledge of the power system. Thus, it is a universal approach and can be deployed in various systems with complicated dynamics. Second, the event detection method relies on classical approaches in QCD, thus is theoretically grounded and can achieve (near-)optimal tradeoff between the probability of false alarm and detection delay. Third, the effectiveness of the proposed approach is validated on a real-world dataset. It outperforms the baseline algorithms, yielding significantly shorter detection delay when the probability of false alarm is reasonably low.

II. System Model

In this paper, we focus on data-driven online event detection in power systems. We consider a discrete-time model, where measurements are collected by certain sensing devices at discrete time points $n = 1, 2, \ldots$. Let $\mathbf{x}_n \in \mathbb{R}^m$ be the m-dimensional measurement vector collected at time n. For example, $\mathbf{x}_n$ could include the voltage, current, etc., measured by a PMU. Due to the inherent temporal correlation existing in the power system, $\mathbf{x}_n$s are not i.i.d. random variables in general. Without loss of generality, we assume that the measurements collected at time n (i.e., $\mathbf{x}_n$) depends on the previous p measurements $\mathbf{x}_{n-p}, \cdots, \mathbf{x}_{n-1}$ through the following equation:

$$\mathbf{x}_n = g(\mathbf{x}_{n-p}, \mathbf{x}_{n-p+1}, \cdots, \mathbf{x}_{n-1}) + \boldsymbol{\epsilon}_n, \tag{1}$$

where $g(\cdot)$ is a function that captures the temporal dependency of the measurements, and $\boldsymbol{\epsilon}_n$ is a random vector that captures other randomness existing in the system. For systems with simple topologies and control space, $g(\cdot)$ may have an explicit form determined by the state transition of the system. However, in general, when the system becomes complicated, $g(\cdot)$ may not admit a clean closed form expression.

We assume $\boldsymbol{\epsilon}_n$ is independent with all previous measurements $\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}$. Besides, depending on whether there exists an event at time n, $\boldsymbol{\epsilon}_n$ may follow different distributions. Specifically, we assume that when the system operates under a normal condition, $\boldsymbol{\epsilon}_n$ is an i.i.d. Gaussian random vector with mean vector to be zero and covariance matrix Σ, i.e., $\boldsymbol{\epsilon}_n \sim \mathcal{N}(\mathbf{0}, \Sigma)$. However, if an event happens at time n, and the system encounters a large disturbance, the distribution of $\boldsymbol{\epsilon}_n$ will be shifted by an *unknown* constant $\boldsymbol{\mu}$, i.e., $\boldsymbol{\epsilon}_n \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$. We assume each event will last a number of time slots, and the objective is to continuously monitors $\mathbf{x}_n$ and detect events in a timely manner once they occur.

The event detection problem can cast into the quickest change detection (QCD) framework. However, compared with standard QCD formulations, there are two main difficulties that we need to address before applying classical QCD approaches to our problem: First, we need to have precise knowledge of $g(\cdot)$ so we can remove the temporal dependencies in the measurements in order to fit the i.i.d. assumption underlying the QCD framework; Second, we need to have distributional information about $\boldsymbol{\epsilon}_n$. In particular, we need to take the *unknown* mean shift $\boldsymbol{\mu}$ after event happens into account.

To address the first challenge, we propose to leverage the expressive power of the LSTM model to approximate $g(\cdot)$. The LSTM model can be trained in an offline fashion. Once trained, it should be able to closely track the temporal evolution of $\mathbf{x}_n$ when the system is stable. For the second challenge, we leverage the generalized likelihood ratio test (GLRT) to estimate the mean shift $\boldsymbol{\mu}$ *online*, and incorporate it to the QCD framework. In the following, we will first introduce the LSTM model and the QCD framework, and then elaborate how to utilize them for quick event detection in power systems.

III. Preliminaries

A. Long Short-Term Memory Model

Long short-term memory (LSTM) model is a recurrent neural network (RNN) first proposed in [15] to overcome the gradient vanishing problem in vanilla RNN [16]. The main idea is to import a memory cell into the RNN structure, which enables the storage and access of information over long periods of time. The memory cell runs straight down the entire chain, and LSTM uses three gates to optionally choose information to add into the memory cell. Such a structure can efficiently overcome the vanishing gradient problem, thus is suitable for time series forecasting problem with long term dependencies.

An LSTM cell is composed of a cell state, an input layer, an input gate, a forget gate and an output gate, as shown in Fig. 1. The historical information is stored in the cell state $\mathbf{c}_n$. The input gate decides which part of the input is worth storing. Such decision is made by considering current input $\mathbf{x}_n$ and previous output $\mathbf{h}_{n-1}$ together as follows: $\mathbf{i}_n = \sigma(\mathbf{W}_i \cdot [\mathbf{h}_{n-1}, \mathbf{x}_n] + \mathbf{b}_i)$, where $\mathbf{W}_i$ and $\mathbf{b}_i$ are the weight matrix and the bias vector of the input gate, respectively, $\sigma()$ is the sigmoid function with output range $(0, 1)$. When $\mathbf{i}_n$ is close to one, it tends to store the information in the cell state. On the other hand, when $\mathbf{i}_n$ is close to zero, it tends to drop the input information. The information selection is then applied by multiplying $\mathbf{i}_n$ with the input vector $\widetilde{\mathbf{c}}_n := \tanh(\mathbf{W}_c \cdot [\mathbf{h}_{n-1}, \mathbf{x}_n] + \mathbf{b}_c)$, where $\mathbf{W}_c$ and $\mathbf{b}_c$ are weight matrix and bias vector of the input layer, respectively. The selected information is denoted as $\mathbf{c}_n^i := \mathbf{i}_n \odot \widetilde{\mathbf{c}}_n$, where $\odot$ is the element-wise multiplication, $\widetilde{\mathbf{c}}_n$ is the input information that combined by current input and previous output, and $\mathbf{c}_n^i$ is the selected information of the input layer that will be stored in the cell state. Similar to the input gate, the forget gate decides which part of the information stored in the cell state should be forgotten. Specifically, $\mathbf{f}_n = \sigma(\mathbf{W}_f \cdot [\mathbf{h}_{n-1}, \mathbf{x}_n] + \mathbf{b}_f)$, $\mathbf{c}_n^f = \mathbf{f}_n \odot \mathbf{c}_{n-1}$, where $\mathbf{W}_f$ and $\mathbf{b}_f$ are weight matrix and bias vector of the forget gate, respectively. Then, the new cell state is $\mathbf{c}_n = \mathbf{c}_n^i + \mathbf{c}_n^f$. The output gate decides which part of information should be output, i.e., $\mathbf{o}_n = \sigma(\mathbf{W}_o \cdot [\mathbf{h}_{n-1}, \mathbf{x}_n] + \mathbf{b}_o)$, $\mathbf{h}_n = \mathbf{o}_n \odot \tanh(\mathbf{c}_n)$, where $\mathbf{W}_o$ and $\mathbf{b}_o$ are weight matrix and bias vector of the forget gate, respectively.

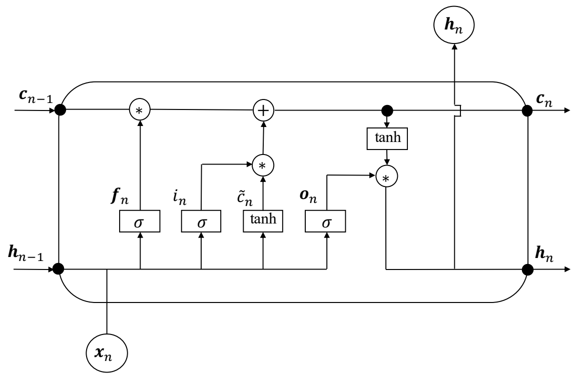

Fig. 1: An LSTM cell.

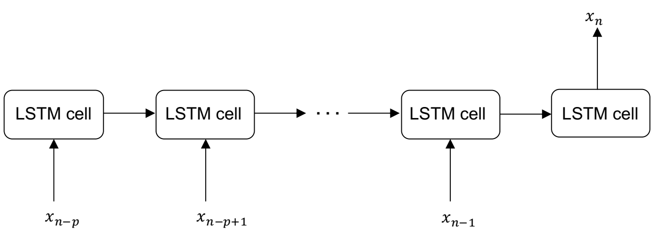

Fig. 2: The LSTM model.

The LSTM model consists of a chain of identical LSTM cells. The input of each LSTM cell corresponds to the measurements of a time step. Thus, the input layer can accommodate historical measurements $\{\mathbf{x}_{n-p}, \cdots, \mathbf{x}_{n-1}\}$ and the future value $\mathbf{x}_n$ can be predicted, as shown in Fig. 2.

B. The Quickest Change Detection Framework

Consider a time series of measurements, denoted as $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \}$. Under the standard QCD framework, it assumes that there exits an unknown change point θ, such that

$$\mathbf{x}_n \sim \begin{cases} f_0(\mathbf{x}_n), & \text{if } n < \theta, \\ f_1(\mathbf{x}_n), & \text{if } n \geq \theta, \end{cases} \tag{2}$$

where $f_0(\cdot)$ and $f_1(\cdot)$ are two different probability distribution functions. The objective of QCD is to obtain a detection procedure δ that maps the observation sequence $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$ to an positive integer $\hat{\theta}$, i.e.,

$$\delta : \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\} \to \{\hat{\theta} : \hat{\theta} \leq n\}, \tag{3}$$

under certain constraints on metrics such as the probability of false alarm (PFA) and the average detection delay (ADD).

Denote the PFA and ADD under detection procedure δ as

$$\text{PFA}(\delta) := \mathbb{P}[\hat{t} < \theta], \quad \text{ADD}(\delta) := \mathbb{E}[\hat{t} - \theta | \hat{t} \geq \theta], \tag{4}$$

where $\hat{t}$ is the time when the detector declares a change and stops to take more measurements. Then, the change detection problem is formally formulated as follows:

$$\min_{\delta} \quad \text{ADD}(\delta), \text{ s.t. PFA}(\delta) \leq \alpha. \tag{5}$$

A typical change point detection method is to monitor the logarithm of the likelihood ratio $f_1(\mathbf{x}_n)/f_0(\mathbf{x}_n)$. A change point can be detected if the two distributions are significantly different. Classical approaches can be classified into the Bayesian framework and the non-Bayesian framework. Under the Bayesian framework, it assumes that the change point is a random variable with known prior distribution, and one of the well known approaches is Shiryaev procedure [17]. In the non-Bayesian framework, it does not assume any distribution on the change point. Classical approaches include the Cumulative Sum (CUSUM) procedure and its variants.

Since it is hard or even impossible to know when an event will happen beforehand in power systems, in this work, we focus on non-Bayesian approaches, especially CUSUM based approaches. The CUSUM procedure was first proposed in [18]. It can be expressed as $\delta_c(h) = \inf\{n : S_n \geq h\}$, where S_n is the cumulative statistic at time n, and h is a pre-defined threshold h. Different threshold h will lead to different tradeoff between PFA and ADD. Denote L_n as the log likelihood ratio based on the observation at time n, i.e., $L_n = \log \frac{f_1(\mathbf{x}_n)}{f_0(\mathbf{x}_n)}$. Then, S_n can be expressed as

$$S_n = \max_{1 \leq k \leq n} \sum_{i=k}^{n} L_i. \tag{6}$$

It has a convenient recursion as follows:

$$S_n = \max(0, S_{n-1}) + L_n, \quad S_0 = 0. \tag{7}$$

It was shown in [19] that CUSUM is asymptotically optimal, i.e., for a given tolerance level on PFA, CUSUM minimizes the worst case detection delay when the average run length goes to infinity.

IV. LSTM-based Event Detection

We note that one critical assumption in the QCD framework is the i.i.d. assumption on the measurements $\mathbf{x}_n$ before and after the change point. However, in power systems, the raw measurements $\mathbf{x}_n$ are in general not i.i.d. in time. In order to fit the QCD framework, in the following, we will first use an LSTM model to approximate the function $g(\cdot)$ in (1) and predict $\mathbf{x}_n$ based on past measurements.

Given the expressive power of LSTM, we expect that the output of LSTM, denoted as $\hat{\mathbf{x}}_n$, will be very close to $g(\mathbf{x}_{n-p}, \ldots, \mathbf{x}_{n-1})$. Thus, the prediction error, denoted as $\mathbf{x}_n - \hat{\mathbf{x}}_n$, will be very close to $\mathbf{e}_n$. Therefore, by utilizing the statistics of the prediction error for detection, the i.i.d. assumption required by the QCD framework can be satisfied.

Specifically, in the following, we will assume $\mathbf{e}_n = \mathbf{x}_n - \hat{\mathbf{x}}_n$ in the following. Then, if an event happens at time θ, we have

$$\mathbf{e}_n \sim \begin{cases} \mathcal{N}(\mathbf{0}, \Sigma), & \text{if } n < \theta, n > \theta + \tau, \\ \mathcal{N}(\boldsymbol{\mu}, \Sigma), & \text{if } n \in [\theta, \theta + \tau], \end{cases} \tag{8}$$

where τ is the duration of the event. We assume τ is sufficiently long so an event can be detected before it ends. Our objective is to detect the change point as quickly as possible based on the prediction error. We note that Σ can be learned from training data, thus is assumed to be known, while $\boldsymbol{\mu}$ is unknown due to the unforeseen events.

A well-known approach to handle unknown parameters in QCD is to incorporate the generalized likelihood ratio test (GLRT) to the CUSUM procedure [19]. The main idea is to use all the past observations to estimate the unknown parameters with the maximum likelihood estimator at each time step. The CUSUM statistic is then calculated using the estimated parameters. Finally, the GLRT-CUSUM statistic is obtained by maximizing the log-likelihood ratio test statistic over all possible change time k as follows:

$$S_n = \max_{1 \leq k \leq n} \sum_{i=k}^{n} \log \frac{f_1(\mathbf{e}_n|\hat{\boldsymbol{\mu}}_k)}{f_0(\mathbf{e}_n)}, \tag{9}$$

where $\hat{\boldsymbol{\mu}}_k$ is the maximum likelihood estimate of $\boldsymbol{\mu}$ assuming a change point happens at time k.

One drawback of GLRT-CUSUM statistic is, it does not have the convenient recursion as CUSUM statistic, thus having a heavy computational complexity. Since it requires to consider all previous observations and all potential parameters at each time step, it may not be very practical to apply it in real-time event detection. To avoid the complicated maximum likelihood estimation, Rao test is adopted to simplify the calculation [20]. Rao test is asymptotically equivalent to GLRT but has a much simpler calculation by taking derivative with respect to the unknown parameters. Under our setting, Rao test statistic can be expressed as follows:

$$\mathcal{R}(\mathbf{e}_n) = \frac{\partial L_n}{\partial \boldsymbol{\mu}}\bigg|_{\boldsymbol{\mu}=\mathbf{0}}^{T} \cdot [\mathbf{J}^{-1}(\boldsymbol{\mu})|_{\boldsymbol{\mu}=\mathbf{0}}] \frac{\partial L_n}{\partial \boldsymbol{\mu}}\bigg|_{\boldsymbol{\mu}=\mathbf{0}}, \tag{10}$$

where $\mathbf{J}(\boldsymbol{\mu})$ is the Fisher information matrix, defined as $\mathbf{J}(\boldsymbol{\mu}) = -\frac{\partial^2 L_n}{\partial \boldsymbol{\mu}^2}$. Then, (9) can be reduced to

$$\bar{S}_n = \max_{1 \leq k \leq n} \sum_{i=k}^{n} \mathcal{R}(\mathbf{e}_i), \tag{11}$$

which is similar to the form of (6) and admits convenient recursion as follows:

$$\bar{S}_n = \max(0, \bar{S}_{n-1}) + \mathcal{R}(\mathbf{e}_n), \quad \bar{S}_0 = 0. \tag{12}$$

With the recursive expression, we can track the approximate CUSUM statistic of $\mathbf{e}_n$ at each time step, and determine whether there is an event or not. If $\bar{S}_n$ reaches the pre-defined threshold h, then an event is declared.

V. Numerical Results

We test the proposed prediction and detection procedure on a dataset collected by Oklahoma Gas & Electric Company (OG&E) [21]. The dataset contains PMU measurements sampled from 285 buses in a 2-minute window, during which a line outage happens. The measurements are collected at a rate of 30 data points per second. In order to illustrate our main idea, in the following, we only use the change of frequency (df/dt) as the variable for event detection. We point out that both the LSTM model and the CUSUM based detection procedure can accommodate multiple variables. Thus, other measured quantities, such as voltage and current, can be exploited together to improve the detection performance potentially.

Under the LSTM model, we use 50 consecutive df/dt measurements to predict the df/dt at the next time step. The model parameters are listed in Table I.

TABLE I: Parameters of the LSTM model.

Time steps	Hidden layer size	Optimizer	Learning rate	Dropout	Epochs
50	100	Adam	1.2e-3	0.1	120

We use the first 850 measurements to train the LSTM model. In the dataset, the event occurs around time index 902, which is indicated by the vertical dashed line in Fig. 3 and Fig. 4. Fig. 3 illustrates the normalized real-time df/dt measurements and the predicted results by the trained LSTM model. We note that the predicted curve closely matches the real measurements before the event, thus the prediction error is small. When an event happens, the measurement is perturbed significantly, leading to more obvious discrepancy between the prediction and the measurement. This corroborates our assumption on the mean shift on $\mathbf{e}_n$ during events. With the trained LSTM model, the approximate CUSUM statistic is tracked in Fig. 4. As we observe, it remains at a low level before the event; Once the event happens, the statistic grows dramatically, triggering the declaration of the event.

Next, we test the event detection performance on all 285 buses. We train an LSTM model for each bus and use the prediction error for event detection under different thresholds. For each threshold h, we record PFA as the percentage of buses experiencing false alarm and record ADD as the average detection delay over all buses. For comparison, we also evaluate the performances of three baseline algorithms. For the first two baseline algorithms, we use a linear regression (LR) model and a standard RNN model for prediction, respectively, and follow the same CUSUM detection procedure as in our proposed approach. The LR model takes 50 consecutive measurements as input and predict the next measurement, and is trained using scikit-learn [22] based on the first 850 measurements. The RNN model shares the same parameters and training process as the LSTM model except that the hidden layer size is set to be 60. For the third baseline algorithm, instead of adopting CUSUM, we simply use the accumulated prediction error from the LSTM model for event detection. Specifically, we monitor the absolute value of the prediction error and compare it with a predefined threshold th_1, which is a scaled standard deviation of the prediction errors during training. Another threshold th_2 is set to declare an event when the sum of the prediction errors that exceed th_1 reaches th_2.

Fig. 5 shows the trade-off between PFA and ADD under those algorithms. As we observe, among those three algorithms that follow the same CUSUM detection procedure, the LSTM based algorithm achieves the best tradeoff. This can be explained as follows: Compared with the LR and RNN models, the LSTM model predicts the measurements more accurately before the event happens, which leads to higher Rao-test statistic when the event happens. Therefore, the corresponding CUSUM statistic reaches the pre-defined threshold more quickly. Besides, with the same LSTM model, the CUSUM based algorithm detects events more quickly than the threshold based detection algorithm, which corroborates the optimality of CUSUM in QCD.

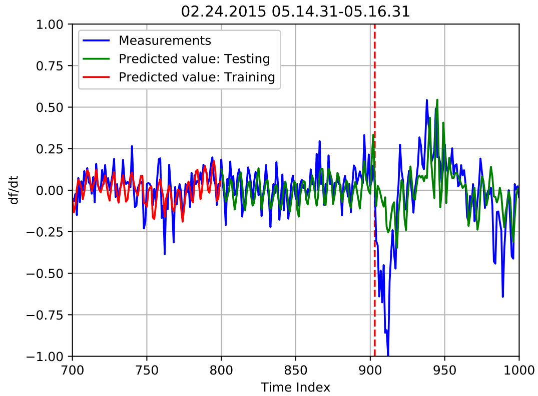

Fig. 3: df/dt prediction.

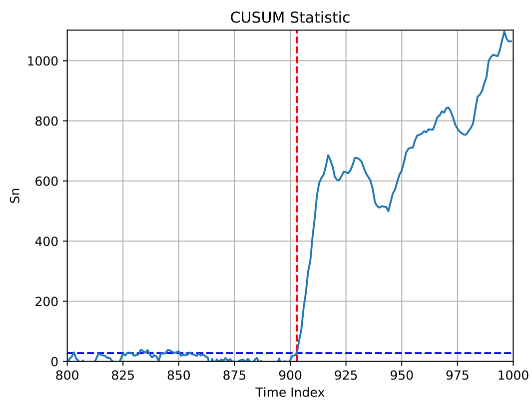

Fig. 4: CUSUM Statistic.

VI. Conclusions

In this paper, we proposed a data-driven approach to detect events in power systems in real time. An LSTM model was adopted to capture the temporal dependencies in the time series measurements. A CUSUM based procedure was then developed by tracking the statistics of the prediction errors. Numerical results based on a real-world PMU dataset were obtained to validate the superior performance of the proposed approach. Future work includes aggregating measurements from multiple buses to detect and locate events, as well as joint training of the LSTM models across buses.

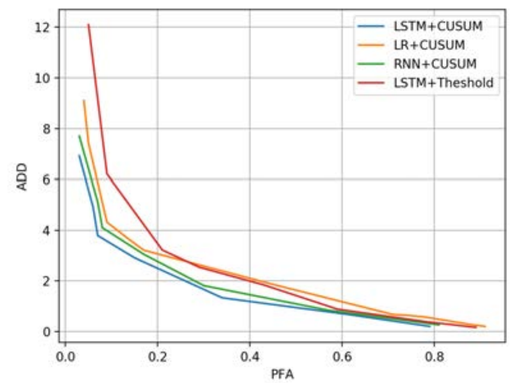

Fig. 5: ADD versus PFA.

References

[1] M. Ceraolo and D. Poli, *Large Power Systems: Structure and Operation*. IEEE, 2014. [Online]. Available: https://ieeexplore.ieee.org/document/6817743

[2] D. Kim, T. Y. Chun, S. Yoon, G. Lee, and Y. Shin, "Wavelet-based event detection method using pmu data," in *IEEE Power Energy Society General Meeting*, July 2017, pp. 1–1.

[3] M. Cui, J. Wang, J. Tan, A. R. Florita, and Y. Zhang, "A novel event detection method using pmu data with high precision," *IEEE Transactions on Power Systems*, vol. 34, no. 1, pp. 454–466, Jan 2019.

[4] J. Y. Zhang and C. M. Bush, "Pmu based islanding detection to improve system operation," in *IEEE Power and Energy Society General Meeting (PESGM)*, July 2016, pp. 1–5.

[5] S. Wen, Y. Wang, Y. Tang, Y. Xu, P. Li, and T. Zhao, "Real-time identification of power fluctuations based on lstm recurrent neural network: A case study on singapore power system," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 5266–5275, Sep. 2019.

[6] H. Ren, Z. Hou, and P. Etingov, "Online anomaly detection using machine learning and HPC for power system synchrophasor measurements," in *IEEE International Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*. IEEE, 2018, pp. 1–5.

[7] G. Gajjar and S. A. Soman, "Auto detection of power system events using wide area frequency measurements," in *National Power Systems Conference (NPSC)*, Dec 2014, pp. 1–6.

[8] Y. C. Chen, T. Banerjee, A. D. Domínguez-García, and V. V. Veeravalli, "Quickest line outage detection and identification," *IEEE Transactions on Power Systems*, vol. 31, no. 1, pp. 749–758, Jan 2016.

[9] Y. Huang, J. Tang, Y. Cheng, H. Li, K. A. Campbell, and Z. Han, "Real-time detection of false data injection in smart grid networks: An adaptive cusum method and analysis," *IEEE Systems Journal*, vol. 10, no. 2, pp. 532–543, June 2016.

[10] Yi Huang, H. Li, K. A. Campbell, and Zhu Han, "Defending false data injection attack on smart grid network using adaptive cusum test," in *45th Annual Conference on Information Sciences and Systems*, March 2011, pp. 1–6.

[11] S. R. Samantaray, A. Samui, and B. Chitti Babu, "S-transform based cumulative sum detector (CUSUM) for islanding detection in distributed generations," in *Joint International Conference on Power Electronics, Drives and Energy Systems*, Dec 2010, pp. 1–6.

[12] X. He, M. Pun, C. . J. Kuo, and Y. Zhao, "A change-point detection approach to power quality monitoring in smart grids," in *IEEE International Conference on Communications Workshops*, May 2010, pp. 1–5.

[13] P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Long short term memory networks for anomaly detection in time series," in *Proceedings*. Presses universitaires de Louvain, 2015, p. 89.

[14] P. Hayton, S. Utete, D. King, S. King, P. Anuzis, and L. Tarassenko, "Static and dynamic novelty detection methods for jet engine health monitoring," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 365, no. 1851, pp. 493–514, 2006.

[15] Maigha and M. L. Crow, "Cost-constrained dynamic optimal electric vehicle charging," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 2, pp. 716–724, Apr. 2017.

[16] S. Wang, S. Bi, Y. Jun, Zhang, and J. Huang, "Electrical Vehicle Charging Station Profit Maximization: Admission, Pricing, and Online Scheduling," *ArXiv e-prints*, 2017.

[17] A. N. Shiryaev, "The problem of the most rapid detection of a disturbance in a stationary process," 1961.

[18] E. S. Page, "Continuous inspection schemes," *Biometrika*, vol. 41, no. 1/2, pp. 100–115, 1954.

[19] G. Lorden, "Procedures for reacting to a change in distribution," *Ann. Math. Statist.*, vol. 42, no. 6, pp. 1897–1908, 12 1971.

[20] T. S. Rao and M. Gabr, "A test for linearity of stationary time series," *Journal of time series analysis*, vol. 1, no. 2, pp. 145–158, 1980.

[21] Oklahoma Gas & Electric Company, https://www.oge.com.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.