

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

# Missing Value Replacement for PMU Data via Deep Learning Model with Magnitude Trend Decoupling

YUANBIN CHENG<sup>1</sup>, (Student Member, IEEE), BRANDON FOGGO<sup>1</sup>, (Member, IEEE), KOJI YAMASHITA<sup>1</sup>, (Member, IEEE), NANPENG YU<sup>1</sup>, (Senior Member, IEEE).

<sup>1</sup>Electrical and Computer Engineering, University of California Riverside, Riverside, CA, 92521 USA (e-mail: [yuchen871@, brandon001@, kyamashi@, and nyu@ece.ucr.edu])

Corresponding author: Nanpeng Yu (e-mail: nyu@ece.ucr.edu).

Disclaimer: this report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

**ABSTRACT** This paper develops a forecasting-based missing value replacement model for Phasor Measurement Unit (PMU) data during power system events. The proposed forecasting model leverages a sequence-to-sequence (Seq2Seq) long short-term memory (LSTM) network with an attention mechanism, which is trained with both pre-event and post-event data. The trained forecasting model is utilized to accurately estimate and recover missing measurements in PMU data. To improve the accuracy of the proposed model, we introduced two novel techniques: (1) integrating a prior knowledge matrix into the attention mechanism that effectively preserves correlations within PMU data, and (2) decoupling the magnitude and trend components of the residual forecast so that such forecasting models are separately trained, which boosts the resistance to the noisy signal. Numerical studies on real-world PMU data collected from the North American electric power grid demonstrate that our proposed model achieves 5% to 30% error reduction compare to the baseline models for all key power grid measurements in both root mean square error and mean absolute error metrics. Furthermore, our model exhibits robust missing data recovery performance even when nearly all of the grid event data is missing.

**INDEX TERMS** Power systems, machine learning, deep learning, missing value replacement, phasor measurement unit, grid disturbance, multivariate time series.

## Nomenclature

$\beta_\tau$	Attention weights at time step $\tau$	$d_\tau$	Hidden state vector at time step $\tau$ in the decoder LSTM network
$\mathbf{X}$	Pre-processed multivariate time-series data	$F_{\text{forecast}}$	Forecasting model
$\mathbf{x}_i$	Observed PMU values at time step $i$	$F_{\text{mag}}$	Magnitude module in the forecasting model
$\mathbf{X}_{i,m}$	Subset time-series of $\mathbf{X}$ start from index $i$ with length $m$	$F_{\text{tre}}$	Trend module in the forecasting model
$\mathbf{y}_\tau$	Observed PMU values at time step $\tau$ in subsequence	$h_t$	Hidden state vector at time step $t$ in the encoder LSTM network
$\mathbf{Y}_t$	Future time-series values after time step $t$	$hb_t$	Backward hidden state at time step $t$ in the encoder LSTM network
$\tau$	Time step of the output (forecast) sequence	$hf_t$	Forward hidden state at time step $t$ in the encoder LSTM network
$b$	The bias of the fully connected layer to generate final forecasting result	$l$	Attention score calculation function
$c_\tau$	Context vector at time step $\tau$	$n$	Number of PMUs in the power system

$P$	Prior knowledge matrix
$R$	Cell function for recurrent neural network
$t$	Time step of the input (past) sequence
$V_a^T$	Second layer weights of the perceptron to calculate attention score
$W$	The weights of the fully connected layer to generate final forecasting result
$W_a$	The weights of the first layer of the perceptron to calculate attention score
$y_\tau$	Forecasted PMU values at time step $\tau$

## I. INTRODUCTION

**P**HASOR Measurement Units (PMUs) are measurement devices used in electric power systems to retrieve time-synchronized, real-time measurements of electrical quantities such as voltage and current phasors (magnitudes and phase angles) at various locations in the power transmission grid. In the past two decades, electric utilities around the world [1], [2] have widely deployed PMUs in the bulk power system to enhance system observability and improve situational awareness for system operators. In the United States alone, more than 2500 PMUs have been deployed in transmission grids, resulting in a significant increase in streaming data available for power grid monitoring. In academia, many researchers have developed PMU data-based algorithms which include power system state estimation [3], [4], event detection (including fault detection) [5], [6], event classification [7]–[10], and offline event analysis [11]. In the power industry, various PMU applications related to situational awareness for high-frequency dynamics, such as renewable energy sources [12], [13], continuous generator model validation [14], asset health monitoring [15]–[17], and linear state estimation [18], [19] have been adopted by Regional Transmission Operators (RTOs) and Independent System Operators (ISOs) in synchrophasor projects. However, there are very few commercial PMU applications, such as eLSE [20], due to the poor PMU data quality and the lack of the grid operators' understanding of the data-driven PMU applications. The poor PMU data quality issue caused by bad data and missing values is one of the most pressing technical challenges for PMU applications to overcome. Missing value replacement is crucial, as data drop as well as latency in PMU data can significantly deteriorate the performance of synchrophasor-based applications, particularly those involving real-time feedback control [21].

The major factor affecting data quality is PMU data loss [22]. Such missing data can be caused by PMU hardware malfunction, GPS-time synchronization issues, and data transmission/communication delays. The current generation of PMU applications merely isolate bad PMU measurements and rarely recover missing PMU measurements. Such data recovery functions are obviously crucial for PMU applications to capitalize on measured data. The most advanced data recovery function is based on model-based estimates and is currently implemented in PMU applications [18]. However, such data recovery functions are used for steady-state PMU measurements only. Applying data recovery functions to

grid event PMU applications is still in the research phase. Although power systems normally operate in the normal condition (steady-state), the PMU data integrity during power system events is more critical to the power system operators because blackouts or brownouts may occur following major grid disturbances. Missing value recovery during the events should get more attention for future PMU applications.

The availability of PMU measurements from across the United States has greatly accelerated the research on machine learning-based PMU data analytics. Many researchers have developed data-driven algorithms to recover missing PMU data. The current approaches for replacing missing values in streaming PMU data can generally be classified into two categories.

The first category of methods for handling missing data in PMUs is matrix completion, which relies on past and currently available PMU measurements to fill in the gaps. [23] fills in the missing PMU data by utilizing other PMUs' dynamic behaviors by averaging the begged multiple linear regression model. [24] performs a tensor decomposition on PMU measurements that have missing data, and converts the decomposed factor matrices back to determine the missing values. [25] introduces an event-decomposition participation method that separates PMU data into steady-state and dynamic components, taking advantage of the low-rank matrix characteristic of streaming PMU data. [26] uses low-rank tensor factorization and subspace selection (known as OnLine Algorithm for PMU data processing, or OLAP) to replace missing values. [27] proposes an OLAP specializing in the temporal aspect of the PMU data that primarily uses past values to fill in missing data, and reference [28] advocates that this version is the most advanced algorithm in the field of PMU missing value replacement. However, these matrix completion-based methods are not able to handle extreme cases where all or a majority of PMUs are out of service for a period of time, such as when GPS signals are out-of-service or malfunctioning.

The second category is the forecasting-based method, which primarily uses historical PMU measurements to fill in the missing data. One of the most significant strengths is that the forecasting-based method can cope with the aforementioned severe conditions, i.e., most or all of the PMUs are out of service. [29] employs the Auto Regressive model that uses the past three observations to infer the current PMU measurements. A Lagrange interpolation method is proposed in [30] that effectively recovers the missing values using a few past observations and the Lagrange polynomial coefficient. [31] proposed a time series-based prediction model that effectively combines Kalman filtering and smoothing algorithms to improve data accuracy. Methods in this category fill in the missing PMU data using the observations from the past few time steps. However, during power system events (e.g., line faults and generator tripping), the PMU measurements often demonstrate abrupt and significant dynamic change. The observations from the past few time steps alone are not sufficient to forecast the event behaviors of hundreds of

milliseconds ahead.

To the best of our knowledge, no data-driven missing value replacement schemes (in the second category) are based on a deep neural network trained with historical power grid event data. This paper employs the forecasting-based scheme and develops a deep learning model that is trained with hundreds of recorded power system events and corresponding PMU data to replace missing values. The proposed model uses a sequence-to-sequence (Seq2Seq) long short-term memory (LSTM) neural network equipped with an attention mechanism, which is well-suited for predicting multivariate time-series data. The model is specifically designed to accurately predict PMU data for 2-8 seconds in the future using pre-vent time-series data of 1 second. Observed short-term correlation patterns among the time-series data inspire us to express these patterns as a prior knowledge matrix and integrate this matrix into the attention mechanism to improve data forecast accuracy. Additionally, we introduce a novel method, which decouples the magnitude and trend components of the data and trains two separate Seq2Seq LSTM models to forecast these components simultaneously. The forecasting model is trained with hundreds of real-world power system events and the corresponding PMU data. The testing results show that our proposed model can provide accurate forecasts and effectively replace missing data even when all PMU measurements are missing simultaneously.

The main contributions of this paper are listed as follows:

- Development of a forecasting-based model with the following two novel techniques to fill the missing measurement in PMU data during the events even when no PMUs are available:
  - Integrate the prior knowledge matrix into the attention mechanism embedded in the Seq2Seq LSTM to improve the data prediction performance.
  - Decouple the PMU data into magnitude and trend components and separately train two deep learning models to better estimate the upward/downward trend of missing PMU data and improve the prediction accuracy.
- Superb prediction accuracy for missing PMU data compared to five baseline deep learning-based missing value replacement models.
- Outstanding prediction accuracy specifically for a large fraction of missing PMU data (over 30%) compared to the state-of-the-art matrix completion-based missing value replacement model.

The rest of this paper is organized as follows: Section II formulates the PMU data forecasting problem and illustrates the overall framework of the proposed method. Section III presents two innovative techniques in the deep learning model. Section IV evaluates the performance of the proposed and baseline models for replacing missing PMU data with a large-scale real-world PMU dataset. Section V concludes this paper.

## II. PROBLEM SETUP AND OVERALL FRAMEWORK

The focus of this section is the formulation of the online PMU data forecasting problem and the introduction of a deep learning-based missing value replacement framework. The proposed model utilizes an attentional Seq2Seq LSTM network, which is enhanced with the incorporation of a prior knowledge matrix and magnitude trend decoupling scheme. The input for the model is the historical time-series data for PMU consisting of four different measurements (real power, reactive power, voltage magnitude, and frequency). The output is a prediction of these same measurements that can be used to replace missing values.

### A. PROBLEM SETUP

The grid-wide sensor system of each interconnection in the U.S. includes a large number of PMUs (hundreds). In this paper, we formulate the replacement of missing values in PMU data as a multivariate time-series forecasting problem.

**Notation 1:**  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T]$  is the pre-processed multivariate PMU time-series data, where  $T$  is the length of the time series  $\mathbf{X}$ ,  $\mathbf{x}_i \in \mathbb{R}^n$  represents the recorded value of different PMUs at the time of  $i$ , and  $n$  is the number of PMUs.

**Notation 2:**  $\mathbf{X}_{i,w} = [\mathbf{x}_i, \mathbf{x}_{i+1}, \dots, \mathbf{x}_{i+w-1}]$  is the expression of a continuous subset time-series of  $\mathbf{X}$  with the length of  $w$  and the start time index of  $i$ , which is used in notation 3.

**Notation 3:**  $\mathbf{X}_{t-m+1,m} = [\mathbf{x}_{t-m+1}, \mathbf{x}_{t-m+2}, \dots, \mathbf{x}_t]$  is the historical time-series values with the fixed length  $m$  (1 second), which is used as the input of the forecasting model.

**Notation 4:**  $\mathbf{Y}_t = [\mathbf{x}_{t+1}, \mathbf{x}_{t+2}, \dots] = [\mathbf{y}_1, \mathbf{y}_2, \dots]$  is the future time-series, which is used as the output of the forecasting model.

The goal of this forecasting task is to predict the future measurement value ( $\mathbf{Y}_t$ ) using past time-series values ( $\mathbf{X}_{t-m+1,m}$ ).  $\mathbf{X}_{t-m+1,m}$  and  $\mathbf{Y}_t$  are matrices with two dimensions that correspond to the spatial and temporal information, i.e., the PMU IDs and timestamps. The output time-series' lengths are 2 seconds for voltage-related events and 8 seconds for frequency-related events.

### B. OVERALL FRAMEWORK

The overall framework of the proposed deep learning-based missing value replacement model is depicted in Fig. 1. The raw dataset undergoes noise removal, outlier detection, and normalization using an Apache Spark Cluster [32]. Then, the pre-processed and normalized training dataset is used to calculate the prior knowledge matrix. The deep learning neural network is constructed based on the Seq2Seq LSTM with an attention mechanism and prior knowledge matrix embedding. The model is trained using the pre-processed PMU data. The deep learning-based model consists of two separately trained neural network modules, the magnitude forecasting module and the trend forecasting module, which are both depicted in Fig. 1 and discussed later (see III-A2). The future PMU data is forecasted by combining the results

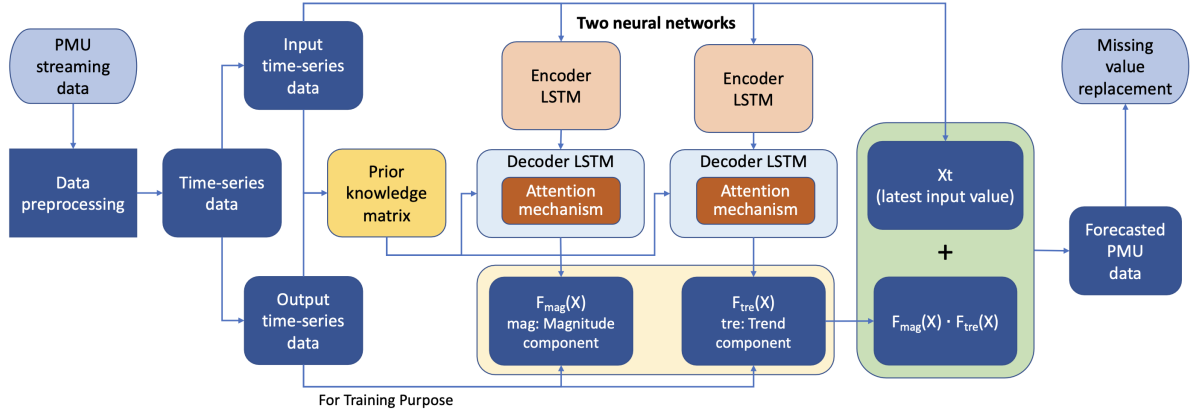


FIGURE 1. Overall framework of the forecasting-based missing value replacement model.

from the aforementioned two modules. Each module will be detailed in the next section.

### III. TECHNICAL METHODS

In this section, we describe the technical methods used in the forecasting model, which tries to predict the residual time series in the future based on the past time series data. First, the time series is separated into the magnitude and trend components, and two separate neural networks are trained to predict each component. The forecast for PMU data is generated by combining the outputs of these two neural networks. Second, we provide a detailed description of the neural network structure of our proposed model with the uniquely designed prior knowledge matrix.

#### A. OVERVIEW OF THE FORECASTING MODEL

##### 1) Forecasting Residual

To predict future PMU data, the proposed model learns the residual mapping and predicts the change (residual) in measurements after the last input measurement instead of directly predict the PMU data time-series. The final prediction is obtained by adding the output of the neural network to the last input measurements, as shown in the following equation:

$$\mathbf{y}_\tau = F_{\text{forecast}}(\mathbf{x}_{1,t}, \mathbf{y}_{1,\tau-1}) + \mathbf{x}_t, \quad (1)$$

where  $\mathbf{y}_\tau$  is the predicted PMU data at the time step of  $\tau$ ,  $F_{\text{forecast}}(\mathbf{x}_{1,t}, \mathbf{y}_{1,\tau-1})$  is the output from the decoder of the LSTM network. The forecasting model  $F_{\text{forecast}}$  only needs to learn the residual mapping between the  $\mathbf{y}_\tau$  and  $\mathbf{x}_t$ .

##### 2) Magnitude-Trend Decoupling

The multivariate time-series data stream from PMU is characterized by noisy signals, which significantly lower the mean directional accuracy (MDA) of its forecasting result. An inaccurate trend in predicting future values can have far more severe consequences than an erroneous magnitude, and the ambient noise in PMU data can significantly reduce the

accuracy of trend predictions in forecasting models. To address this issue, this paper proposes breaking down the forecasting problem into two simpler sub-problems: magnitude forecasting and trend forecasting. This can be accomplished by leveraging two distinct models, one dedicated to capturing magnitude, and the other specifically designed to capture trend. We decouple the magnitude and trend components of the residual time-series data using the following equation:

$$\mathbf{y}_\tau = F_{\text{mag}}(\mathbf{x}_{1:t}, \mathbf{y}_{1:\tau-1}) \times F_{\text{tre}}(\mathbf{x}_{1:t}, \mathbf{y}_{1,\tau-1}) + \mathbf{x}_t, \quad (2)$$

where  $F_{\text{tre}}(\mathbf{x}_{1:t}, \mathbf{y}_{1,\tau-1})$  predicts the future trend (i.e., trend of the time-series data). Each forecasted value equals to +1 (increasing) or -1 (decreasing).  $F_{\text{mag}}(\mathbf{x}_{1:t}, \mathbf{y}_{1,\tau-1})$  predicts the magnitude of the future PMU data point, i.e., a positive value indicating the absolute value of the change in the measurement.

In this study, we start from a Seq2Seq LSTM model with an attention mechanism to forecast PMU data, subsequently utilizing the model to impute missing values. This model is used by two separate modules for trend and magnitude, and the outputs from these modules are combined to generate the final prediction. The architecture of the neural network will be discussed in further detail in the following subsection.

#### B. NEURAL NETWORK ARCHITECTURE

The neural network architecture of the Seq2Seq LSTM equipped with the attention mechanism is illustrated in Fig. 2. Since the network design is based on the Seq2Seq structure, it has no limit on the input and output data length.

This model consists of two key components: the encoder and the decoder, and the decoder is equipped with an attention layer and prior knowledge matrix. Both the encoder and the decoder have a bi-directional recurrent neural network (RNN) structure. The attention mechanism with the prior knowledge matrix in the decoder extracts a weighted feature to model the temporal dependency between the input and output time series. How the above two components extract

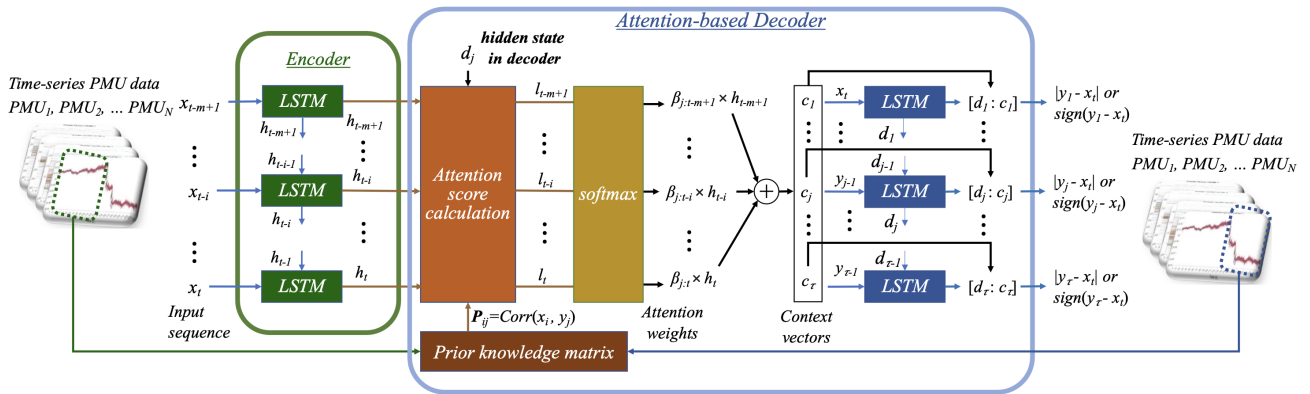


FIGURE 2. Neural network architecture to predict missing PMU values.

and learn the temporal correlations in the time-series data is discussed below.

### 1) Encoder

The encoder is built on an RNN, which can learn inherent features in past time-series values to predict the future. Given a time-series data,  $[x_{t-m+1}, x_{t-m+2}, \dots, x_t]$ , where  $x_i \in \mathbb{R}^n$ . An RNN generally defines a recurrent function,  $R$ , and calculates the hidden state,  $h_t \in \mathbb{R}^m$ , for each time step,  $t$ , as:

$$h_t = R(h_{t-1}, x_t), \quad (3)$$

where the function,  $R$ , depends on the type of RNN cell (a loop-shaped neural network structure in hidden layers) is used.

Backpropagation through time allows the hidden states to capture temporal features during training, but it can suffer from gradient vanishing or exploding over longer timescales, which can significantly degrade the model performance. To cope with this issue, the LSTM network was introduced as a variant of the basic RNN. LSTM units consist of a memory cell and three controlled gates, as described in [33]. Therefore, we chose to use an LSTM network in our forecasting model to effectively capture time dependencies and extract features for each time step in the input sequence.

Additionally, the proposed model utilizes a bi-directional LSTM network to capture temporal features at each time step by incorporating both past time-series values,  $hf_t$ , and future values,  $hb_t$ . This allows for a more comprehensive understanding of the temporal dynamics within the time-series data set.

$$\text{forward: } hf_t = \text{LSTM}(x_t, hf_{t-1}) \quad (4)$$

$$\text{backward: } hb_t = \text{LSTM}(x_t, hb_{t+1}) \quad (5)$$

$$h_t = [hf_t; hb_t] \quad (6)$$

As shown in (4) and (5), the forward and backward LSTMs in the encoder network simultaneously process the time-

series data in opposite directions. The forward LSTM processes data from  $t = 1$  to  $T$  using a forward hidden layer, while the backward LSTM processes data from  $t = T$  to 1 using a backward hidden layer. The final hidden state for each time step, denoted as  $h_t$ , is obtained by concatenating the output of the forward and backward hidden layers at time step  $t$ . The time-series data is fed to the encoder as the input. The output is a time-series of hidden states  $h_t$  that are further processed in the decoder layer's attention mechanism to calculate the attention distribution for each time step in the output time-series data.

### 2) Attention-based Decoder

The decoder neural network is also built based on a bi-directional LSTM network. To prevent squashing all the time-series data into a fixed-length vector (the last hidden state from the encoder), [34] and [35] proposed the attention mechanism, which enables the forecasting model to selectively focus on specific aspects of the input time-series data. Because all the input hidden states are equally and fully leveraged, the model's performance is not affected by the hidden states squashing. Thus, our proposed forecasting model capitalizes on the attention mechanism to propagate predominant features in the time-series data from the encoder to the decoder. Moreover, a prior knowledge matrix is integrated into the attention mechanism as fixed parameters to sharpen the attention.

The calculation of the forecast value of each time step in the decoder is based on the LSTM hidden state at the current time step,  $d_\tau$ , all the hidden states of the input time-series data,  $h_{1..t}$ , the latest previous forecasting result,  $y_{\tau-1}$ , and a pre-calculated prior knowledge matrix,  $\mathbf{P}$ :

$$y_\tau = F(d_\tau, y_{\tau-1}, h_{1..t}, \mathbf{P}) \quad (7)$$

Each element of the prior knowledge matrix  $\mathbf{P}_{i,j}$  is calculated by the average Pearson correlation coefficient between the measurements,  $x_i$  and  $y_j$ . The prior knowledge matrix

directly captures the temporal correlations between various time steps of the input and output time-series.

$$\mathbf{P}_{i,j} = \text{Average}_{\mathbf{x}, \mathbf{y} \in \text{Training}} (\text{Corr}(\mathbf{x}_i, \mathbf{y}_j)) \quad (8)$$

The attention mechanism is implemented by a two-layer perceptron, followed by an LSTM network in the decoder. For each encoder output  $h_i$ , its attention score,  $l(d_\tau, h_i)$ , is computed as:

$$l(d_\tau, h_i) = \mathbf{P}_{\tau,i} \times V_a^T \tanh(W_a[d_\tau; h_i]), \quad (9)$$

where  $d_\tau$  refers to the decoder's hidden state at time step  $\tau$ ,  $h_i$  denotes the hidden state of the input time-series from the encoder at time step  $i$ .  $V_a^T$  and  $W_a$  are the weights of the two-layer perceptron in the attention mechanism.

The attention weight,  $\beta_\tau$ , is calculated by applying the softmax normalization for the attention scores  $l$ :

$$\beta_\tau(d_\tau, h_i) = \text{softmax}(l(d_\tau, h_i)) = \frac{\exp(l(d_\tau, h_i))}{\sum_{i=1}^t \exp(l(d_\tau, h_i))} \quad (10)$$

The attention weight  $\beta_\tau(d_\tau, h_i)$  represents the degree of correlation between the  $\tau$ -th output and the input at the time of  $i$ . To synthesize the temporal features across all the input time steps, the context vector,  $c_\tau$  is used. It is calculated as the weighted sum of all encoder outputs as shown below:

$$c_\tau = \sum_{i=1}^t \beta_\tau(d_\tau, h_i) \cdot h_i \quad (11)$$

Each context vector at the time of  $\tau$ ,  $c_\tau$ , combines the encoder's hidden states at every time step. It should be noted that the context vector contains the feature from all previous time-series data.

Finally, the concatenation of the context vector  $c_\tau$  and the hidden state  $d_\tau$  form the feature vector for the output series, and a fully connected layer is used to get the forecast result for time step  $\tau$ . This is shown in (12).

$$\mathbf{y}_\tau = W \times [d_\tau; c_\tau] + b, \quad (12)$$

where  $W$  and  $b$  are the weights and bias of the fully connected layer.

#### IV. NUMERICAL STUDY

This section assesses the performance of the proposed forecasting model by comparing it to five forecasting-based models and the matrix completion model. The prediction accuracy of the forecasting-based models is evaluated using two indicators: root mean square error (RMSE) and mean absolute error (MAE). These indicators are chosen due to their suitability for evaluating voltage and frequency behaviors during voltage and frequency events. The prediction accuracy of the matrix completion model is evaluated using the mean absolute percentage error (MAPE) when a significant portion of PMU data is missing. The contribution of the prior knowledge matrix is also analyzed by visualizing the attention weights.

#### A. DATA SOURCE AND DATA PRE-PROCESSING

##### 1) Data Source

The data for this study was collected from the PMUs located in the Western Electric Coordination Council (WECC) over a two-year period (2016-2017) at a sampling frequency of 30 Hz. The original data set consists of 42 PMUs, totaling over 6 Terabytes. However, one PMU was excluded from the analysis due to being mostly out of service, resulting in a final sample of 41 PMUs.

In addition, historical operator logs from RTOs and ISOs provided 958 power system event labels, which were divided into voltage-related (caused by system faults at transformers, lines, and buses) and frequency-related (generating unit trip) events.

##### 2) Data Pre-processing

The data cleaning process involves the removal of outliers from the dataset, filling in missing data using the method described in [25], calculating the real and reactive power, and organizing the streaming PMU data into a three-dimensional tensor. The three dimensions correspond to the timestamp, PMU ID, and measurement channel. This process is performed using a data pipeline implemented with a five-node Apache Spark and Hadoop cluster, which allows for the efficient handling of large-scale data. In Fig. 3, we present examples of voltage-related and frequency-related events to demonstrate the behavior of four distinct measurements taken during the event after the data pre-processing. The start time of the event is at 9 seconds. These event labels included the timing and category/type of the event.

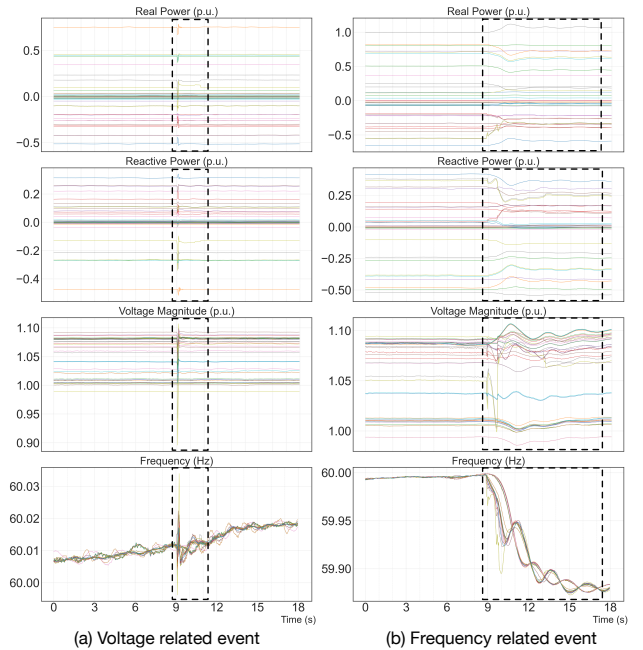


FIGURE 3. PMU data for example frequency- and voltage-related events (with forecast data range marked by the dotted area). Each curve on the plot corresponds to data collected from a unique PMU, with the colors chosen to provide clear differentiation between them.

### 3) Dataset and Case Study Setup

After cleaning the data, we pre-process it based on the event types. During the model training process, we use different forecasting lengths depending on the event type, as shown in Fig. 4. This figure illustrates the period for data extraction of each event in the dataset. The event period sequence that we extracted starts one second before the event label time stamp and continues for either two or eight seconds after the event label time stamp, depending on the event type.

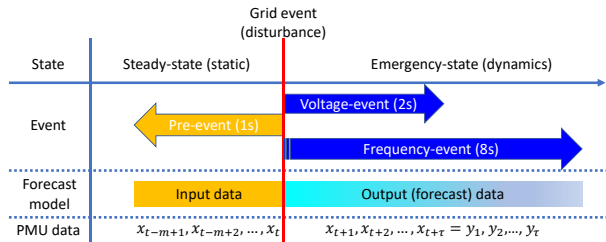


FIGURE 4. Time-series PMU data to be trained and predicted.

The PMU data are extracted from all 958 labeled events. 60% of the events are selected as the training dataset, 20% of the events are used as the validation dataset for the hyperparameter tuning, and the remaining 20% are to evaluate the model performance.

The model parameters are learned with the training dataset, while the early-stopping function is used to monitor the loss on the validation dataset to avoid overfitting. RMSE is adopted as the evaluation metric for the proposed model, which is given by:

$$RMSE = \sqrt{\frac{1}{n} \sum_{j=1}^n (y_j - \hat{y}_j)^2}, \quad (13)$$

where  $\hat{y}$  is the forecasted data, while  $y$  is the measured data.

The training process uses the Adam optimizer and the learning rate is selected to be 0.001. A Linux server with 4 Nvidia RTX 2080 Ti GPU is used to train the proposed and baseline machine learning models.

### B. BASELINE METHODS

The present study investigates the performance of five baseline forecasting models: linear regression, fully connected neural network, convolutional neural network (CNN), Seq2Seq LSTM model, and the Seq2Seq LSTM model with Luong attention. The parameters for each model are provided in Table 1, including the range of hyperparameters used in training. All models are trained using the Adam optimizer and the RMSE loss function on the same dataset. The specific characteristics and performance of each model will be discussed in the subsequent subsections.

#### 1) Linear Regression

The linear regression model assumes the output measurements have a linear relationship with the input subsequence.

#### 2) Fully Connected Neural Network

This model uses a multi-layer fully connected neural network to model the relationship between the input and the output subsequences. This neural network model consists of 3 layers of fully connected neurons.

#### 3) Convolutional Neural Network

This model uses CNN to perform the forecasting task. This neural network model comprises five layers with a sliding filter size of  $5 \times 5$ , and 64 filters for each internal layer.

#### 4) Seq2Seq LSTM Model

This model uses the original Seq2Seq LSTM neural network to conduct the forecasting task. Both the encoder and decoder are two-layer Bi-LSTM models. The number of neurons for every layer of the LSTM model is 128.

#### 5) Attentional Seq2Seq LSTM Model

This model appends the Luong attention mechanism to the above mentioned Seq2Seq LSTM model.

### C. PERFORMANCE INDICATORS

Two evaluation metrics are used to compare the prediction performance of the five baseline models and our proposed forecasting model. RSME emphasizes large errors. On the other hand, MAE equally treats the individual errors, which means it has weaker weights for outliers compared to the RMSE. Note that voltage events tend to contain abrupt changes in the voltage signal, while frequency events generally illustrate gradual changes in the system frequency.

$$MAE = \frac{1}{n} \sum_{j=1}^n |y_j - \hat{y}_j|, \quad (14)$$

where  $\hat{y}$  is the forecasted data, while  $y$  is the measured data.

### D. PMU DATA FORECASTING ACCURACY

The performance metrics of various PMU data prediction algorithms are presented in Table 2. The abbreviation ‘‘PKM’’ represents the ‘‘prior knowledge matrix’’ while ‘‘MTD’’ stands for ‘‘magnitude-trend decoupling.’’

As shown in Table 2, the Seq2Seq LSTM algorithm demonstrates superior forecasting performance for all measurement types (active power, reactive power, voltage, frequency) compared to linear regression, fully connected neural networks, and CNN. This is likely due to the LSTM’s ability to model temporal correlations in the PMU time-series effectively. The incorporation of the attention mechanism in the ATT-Seq2Seq-LSTM model leads to a reduction in the forecasting error by enabling the model to better capture the temporal dependencies between the input and output time-series.

The integration of the prior knowledge matrix results in a reduction of prediction error across all four measurement types, particularly for real and reactive power. The application of the magnitude and trend decoupling technique also enhances the expressive power of the forecasting model, which

TABLE 1. Specifications of Baseline Forecasting-based Models

	Linear regression (Linear)	Fully connected neural network (FC NN)	Convolutional neural network (CNN)	Seq2Seq LSTM (Seq2Seq-LSTM)	Seq2Seq LSTM with attention mechanism (ATT-Seq2Seq-LSTM)
Optimizer	Adam	Adam	Adam	Adam	Adam
Learning rate	0.001	0.001	0.001	0.001	0.001
Loss function	RMSE	RMSE	RMSE	RMSE	RMSE
Hidden layer	0	3 (2-8)	5 (2-8)	2 (encoder) (1-4) 2 (decoder) (1-4)	2 (encoder) (1-4) 2 (decoder) (1-4)
Number of hidden neurons	0	128 (32, 64, 128)	64 (32, 64, 128)	128 (32, 64, 128)	128 (32, 64, 128)
Attention algorithm	No	No	No	No	Yes (Luong attention mechanism)

The numbers enclosed in parentheses were used as candidates for hyperparameter tuning, which is the process of selecting the best set of hyperparameters for each forecasting-based model.

TABLE 2. Prediction Errors for Baseline Models and Proposed Forecast Model

Model	Measurement type	Voltage Event		Frequency Event	
		RMSE	MAE	RMSE	MAE
Linear	Real Power (p.u.)	0.00962	0.00589	0.00698	0.00335
	Reactive Power (p.u.)	0.0177	0.00970	0.00722	0.00399
	Voltage (p.u.)	0.0044	0.0020	0.0013	0.00093
	Frequency (Hz)	0.0023	0.0015	0.0028	0.0021
Fully-connected Neural Network (FC NN)	Real Power (p.u.)	0.00493	0.00295	0.00395	0.00246
	Reactive Power (p.u.)	0.00432	0.00205	0.00246	0.00142
	Voltage (p.u.)	0.0028	0.00093	0.00053	0.00033
	Frequency (Hz)	0.0015	0.00095	0.0012	0.0009
Convolutional Neural Network (CNN)	Real Power (p.u.)	0.00471	0.00252	0.00430	0.00263
	Reactive Power (p.u.)	0.00367	0.00181	0.00249	0.00143
	Voltage (p.u.)	0.0026	0.00089	0.00044	0.00027
	Frequency (Hz)	0.0012	0.00069	0.0011	0.00082
Seq2Seq LSTM	Real Power (p.u.)	0.00347	0.00187	0.00366	0.00231
	Reactive Power (p.u.)	0.00340	0.00147	0.00151	0.00086
	Voltage (p.u.)	0.0019	0.00067	0.00041	0.00025
	Frequency (Hz)	0.0010	0.00059	0.00065	0.00045
ATT-Seq2Seq-LSTM	Real Power (p.u.)	0.00320	0.00173	0.00359	0.00229
	Reactive Power (p.u.)	0.00333	0.00140	0.00148	0.00085
	Voltage (p.u.)	0.0017	0.00062	0.00040	0.00024
	Frequency (Hz)	0.00094	0.00052	0.00061	0.00042
ATT-Seq2Seq-LSTM-PKM	Real Power (p.u.)	0.00302	0.00168	0.00329	0.00211
	Reactive Power (p.u.)	0.00304	0.00134	0.00143	0.00082
	Voltage (p.u.)	0.0016	0.00058	0.00039	0.00022
	Frequency (Hz)	0.00088	0.00051	0.00060	0.00041
ATT-Seq2Seq-LSTM-MTD	Real Power (p.u.)	0.00283	0.00159	0.00259	0.00165
	Reactive Power (p.u.)	0.00286	0.00127	0.00141	0.00082
	Voltage (p.u.)	0.0014	0.00054	0.00037	0.00021
	Frequency (Hz)	0.00086	0.00049	0.00059	0.00039
ATT-Seq2Seq-LSTM-PKM-MTD (Proposed Model)	Real Power (p.u.)	0.00275	0.00155	0.00247	0.00152
	Reactive Power (p.u.)	0.00280	0.00122	0.00139	0.00081
	Voltage (p.u.)	0.0014	0.00054	0.00036	0.00021
	Frequency (Hz)	0.00083	0.00048	0.00059	0.00039

Base complex power for real power and reactive power is 1000 MVA.

led to significant improvements in PMU data forecasting performance. In the following subsection, we will examine the ways in which the proposed techniques contribute to the success of the proposed forecasting model.

Note that the RMSE is larger than the MAE for most of the measurement types in Table 2. This occurs when large errors appear in the prediction results. This is mostly due to the fact that PMU time-series include transient signals with significant noise. As shown in the Table, our proposed forecasting model significantly lowers the prediction error in voltage signals during voltage events. During frequency events, our proposed forecasting model significantly lowers

the active power prediction error. This is because voltage and active power signals are the most predominant signals in the voltage and frequency events respectively.

Fig. 5 shows that the MTD is the major contributor to improving the forecasting model accuracy for voltage events. On the other hand, not only the MTD but also PKM contributes to improving the forecasting accuracy for frequency events, especially the active power signals. Thus, both the PKM and MTD effectively reduce the error of the proposed forecasting model for both event types.



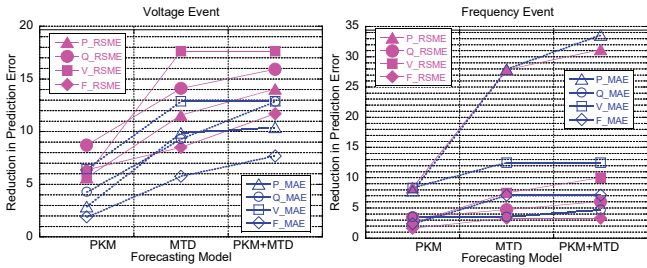


FIGURE 5. Percentage reduction in prediction error by PKM and MTD from Att-Seq2Seq-LSTM.

TABLE 3. Prediction Error of Magnitude Component and Forecast Accuracy of Trend Component

Voltage Events	Magnitude (RMSE)	Trend (Accuracy)
Att-Seq2Seq-LSTM	0.00832	60.19%
Att-Seq2Seq-LSTM-MTD	0.00772	81.67%
Att-Seq2Seq-LSTM-PKM	0.00784	61.42%
Att-Seq2Seq-LSTM-PKM-MTD	0.00732	82.32%
Frequency Events	Magnitude (RMSE)	Trend (Accuracy)
Att-Seq2Seq-LSTM	0.00578	70.20%
Att-Seq2Seq-LSTM-MTD	0.00473	82.59%
Att-Seq2Seq-LSTM-PKM	0.00534	71.59%
Att-Seq2Seq-LSTM-PKM-MTD	0.00443	83.66%

**E. PREDICTION ERROR FOR THE MAGNITUDE COMPONENT AND FORECASTING ACCURACY FOR THE TREND COMPONENT**

Table 3 shows the prediction error for the magnitude component and the forecasting accuracy for the trend component of the PMU data for the proposed and baseline methods. The prediction error for the magnitude component is measured by RMSE and the forecasting accuracy for the trend component is quantified by the accuracy, i.e., up/down binary agreement rate. The RMSE and accuracy reported in the table are the average results across the four measurement types. As shown in Table 3, the proposed decoupling technique reduces the magnitude prediction error and increases the trend prediction accuracy, which significantly contributes to the reduction in overall forecasting error. It can also be observed that the proposed forecasting model exhibits higher accuracy in predicting the trend for frequency-related events than voltage-related events. This is due to the fact that the signals in the frequency-related events are less noisy.

Although the prior knowledge matrix technique also improved the accuracy of the trend prediction and reduced the error of the magnitude prediction, this improvement is limited in comparison to the improvement achieved through the use of the magnitude trend decoupling technique. Thus, using two neural networks to separately predict the magnitude and trend components of the PMU data is the key innovation to effectively and accurately learn the dynamic behavior in the power system.

**F. VISUALIZATION OF ATTENTION WEIGHTS**

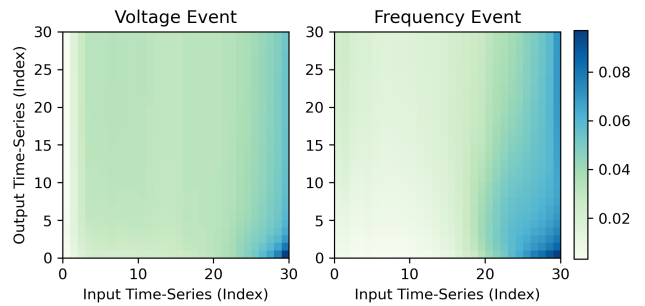


FIGURE 6. Attention weights between different time steps of forecasted subsequence and historical data (without prior knowledge matrix integration).

The attention weights between the input and output time-series of the forecasting model at each time step are illustrated in Fig. 6. These attention weights are calculated without considering the prior knowledge matrix. The input consists of the subsequence that starts at one second before the event label, while the output corresponds to the forecasted PMU data. Higher attention weights between the input and output are indicated by blue colors, while weaker weights are indicated by green colors. As shown in the figure, the attention weights are much higher between the last few time steps of the input time-series and the first few time steps of the output time-series. In other words, the attention mechanism focuses more on the final few input data points when predicting future PMU data. Note that the Seq2Seq LSTM model without the attention mechanism relies solely on the final input time step’s hidden state of the encoder to make predictions about the future PMU data by the decoder. However, the information about the previous time steps of the input sequence may not be fully captured by the final time step’s hidden state, which leads to performance deterioration.

**G. IMPACTS OF PRIOR KNOWLEDGE MATRIX ON ATTENTION WEIGHTS**

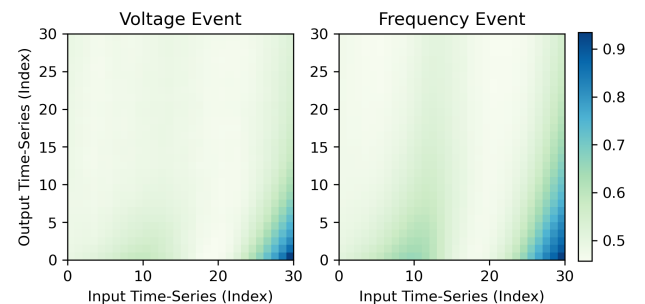


FIGURE 7. Prior knowledge matrix weights between forecasted PMU data and input time-series.

The heat map for the prior knowledge matrix between the input and output time-series are illustrated in Fig. 7. The blue and green colors indicate a strong and weak correlation, respectively. In general, stronger correlations can be observed

between the final few time steps of the input time-series and the first few time steps of the output time-series. This pattern is quite similar to that shown in Fig. 6.

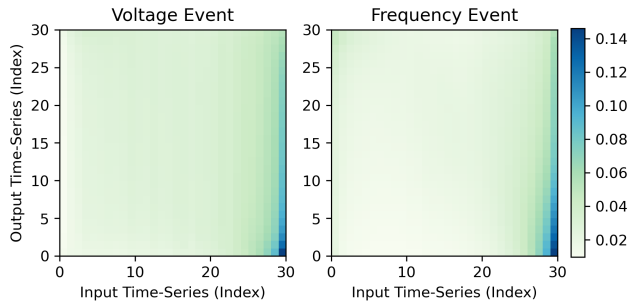


FIGURE 8. Attention weights between different time steps of forecasted subsequence and historical data (with prior knowledge matrix integration).

The impacts of incorporating the prior knowledge matrix on the attention weights are demonstrated in Fig. 8. Here the attention weights are calculated by the model that integrates the prior knowledge matrix with the attention mechanism. By comparing Figs. 6 and 8, it is evident that the incorporation of the prior knowledge matrix in the attention mechanism leads to an increased focus on the final few time steps of the input sequence for predicting the first few time steps of the output time-series. The combination of the attention mechanism and the prior knowledge matrix, therefore, highlights the importance of the final few time steps in the input time-series. The improved forecasting performance demonstrated in Table 2 further supports the significant role of the prior knowledge matrix in the attention mechanism.

### H. MISSING PMU DATA REPLACEMENT PERFORMANCE

This subsection compares the missing value replacement performance of the proposed PMU data forecasting model and a state-of-the-art matrix-completion algorithm, OLAP. The experiments are conducted with missing data ratios ranging from ratio from 10% to 90%.

#### 1) Case Study Setup

For each event and PMU time-series pair, it is assumed that the input data is complete and  $p\%$  of the output time series' first time step data are missing. The proposed and baseline models both estimate missing values. The averaged mean absolute percentage error (MAPE) for the estimated missing data points in all testing events is used as the performance metric.

#### 2) Baseline Method

The state-of-the-art matrix completion-based method, OLAP [26] is used as the baseline method for missing value replacement. OLAP uses singular value decomposition to process the PMU data matrix and identifies the best linear combination of left singular vectors to fill in the missing PMU data.

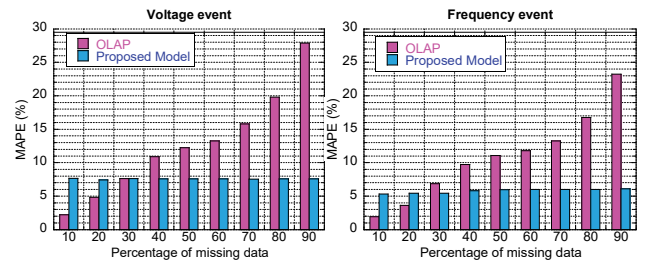


FIGURE 9. Missing value replacement error with respect to percentage of missing data.

#### 3) Missing Value Replacement Performance Comparison

Fig. 9 shows the missing value replacement algorithms' MAPE under different missing data percentages. The MAPE is calculated by averaging over all event labels and four measurement channels for voltage and frequency events. It can be observed that when the percentage of missing value is 30% or lower, OLAP achieves lower MAPE compared to the proposed forecasting-based model. This is not surprising because OLAP capitalizes on the other PMU data at the time when some PMUs have missing data. However, when the percentage of missing data is above 30%, our proposed model significantly outperforms OLAP. The comparative advantage is more pronounced when the missing data percentage is higher. The reason why the MAPE of our proposed model is nearly flat with the percentage of missing data is that the forecasting-based model does not leverage the other PMUs' measurements at the time step when PMUs have missing values.

Finally, the proposed forecasting-based model can be utilized to replace missing PMU values in the most severe event that is when all PMUs are offline due to GPS satellite malfunction. In this case, the baseline method, OLAP, won't be able to estimate any missing values.

### V. CONCLUSION

This paper proposes a deep learning-based forecasting algorithm to fill in missing PMU data during power system events. The proposed deep learning model is trained using PMU data from hundreds of real-world voltage-related and frequency-related events. The proposed model is built on top of a Seq2Seq LSTM model with attention mechanism. The two innovative techniques, integrating the prior knowledge matrix into the attention mechanism and decoupling the PMU data into the magnitude and trend components, successfully improve the prediction accuracy and reduce the forecasting error.

The missing PMU data replacement performance of the proposed and baseline deep learning and matrix completion-based methods are evaluated using real-world PMU data during system events in WECC. The testing results show that our proposed model outperforms the baseline deep learning models in terms of prediction accuracy. When the missing value percentage is higher than 30%, our proposed model achieves much lower missing data estimation error than the

state-of-the-art matrix completion-based approach. Further in-depth numerical study results quantify the impacts of the two innovative techniques (prior knowledge matrix and magnitude-trend decoupling) on the proposed forecasting model's accuracy. These two unique techniques helped reduce the RMSE of the proposed model by 10-20% compared to the best baseline method.

In future work, it would be interesting to explore the use of a hybrid forecasting model that combines the current model with another method, such as the matrix completion method, to better handle a mixture of voltage and frequency events. Additionally, extending the scope of event types to include power swing oscillation and converter-driven oscillation would lead to a more comprehensive approach to missing value replacement in the power system. It should be noted that the grid topology is not disclosed, and only PMU measurements are available for these case studies. If grid topology is available, other advanced neural networks, such as graph convolutional networks, can be promising to improve the missing value imputation performance.

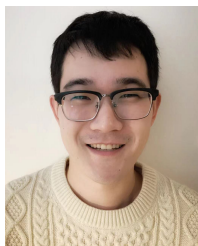
## ACKNOWLEDGMENT

This material is based upon work supported by the Department of Energy under Award Number DE-OE0000916.

## REFERENCES

- [1] A. Monti, C. Muscas, and F. Ponci, *Phasor measurement units and wide area monitoring systems*. Academic Press, 2016.
- [2] A. G. Phadke and T. Bi, "Phasor measurement units, WAMS, and their applications in protection and control of power systems," *J. Mod. Power Syst. Clean Energy*, vol. 6, no. 4, pp. 619–629, Jul. 2018.
- [3] J. Zhao, G. Zhang, K. Das, G. N. Korres, N. M. Manousakis, A. K. Sinha, and Z. He, "Power system real-time monitoring by using PMU-based robust state estimation method," *IEEE Trans. Smart Grid*, vol. 7, no. 1, pp. 300–309, Jan. 2016.
- [4] F. Aminifard, M. Shahidehpour, M. Fotuhi-Firuzabad, and S. Kamalinia, "Power system dynamic state estimation with synchronized phasor measurements," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 2, pp. 352–363, Feb. 2014.
- [5] M. Alqudah, M. Pavlovski, T. Dokic, M. Kezunovic, Y. Hu, and Z. Obradovic, "Fault detection utilizing convolution neural network on timeseries synchrophasor data from phasor measurement units," *IEEE Trans. Power Syst.*, vol. 37, no. 5, pp. 3434–3442, Sep. 2022.
- [6] A. A. Hai, T. Dokic, M. Pavlovski, T. Mohamed, D. Saranovic, M. Alqudah, M. Kezunovic, and Z. Obradovic, "Transfer learning for event detection from PMU measurements with scarce labels," *IEEE Access*, vol. 9, pp. 127 420–127 432, Sep. 2021.
- [7] J. Shi, B. Foggo, and N. Yu, "Power system event identification based on deep neural network with information loading," *IEEE Trans. Power Syst.*, vol. 36, no. 6, pp. 5622–5632, Nov. 2021.
- [8] M. Pavlovski, M. Alqudah, T. Dokic, A. A. Hai, M. Kezunovic, and Z. Obradovic, "Hierarchical convolutional neural networks for event classification on PMU measurements," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–13, Sep. 2021.
- [9] Y. Yuan, Y. Guo, K. Dehghanpour, Z. Wang, and Y. Wang, "Learning-based real-time event identification using rich real PMU data," *IEEE Trans. Power Syst.*, vol. 36, no. 6, pp. 5044–5055, Nov. 2021.
- [10] Y. Yuan, Z. Wang, and Y. Wang, "Learning latent interactions for event classification via graph neural networks and PMU data," *IEEE Trans. Power Syst.*, vol. 38, no. 1, pp. 617–629, Jan. 2023.
- [11] M. Ghorbaniparvar, N. Zhou, X. Li, D. J. Trudnowski, and R. Xie, "A forecasting-residual spectrum analysis method for distinguishing forced and natural oscillations," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 493–502, Jan. 2019.
- [12] P. Gravois, "Synchrophasor based oscillation detection in ERCOT detection in ERCOT operations," in *NASPI Work Group Meeting and Vendor Show*, Sep. 2017, Accessed: Dec. 30, 2022. [Online]. Available: [https://www.naspi.org/sites/default/files/2017-10/03\\_ercot\\_gravois\\_synchrophasor\\_oscillation\\_detection\\_20170926.pdf](https://www.naspi.org/sites/default/files/2017-10/03_ercot_gravois_synchrophasor_oscillation_detection_20170926.pdf)
- [13] J. Hariharan, "Odessa 2 - lessons learned," in *NERC-NASPI IBER Workshop*, Sep. 2022, Accessed: Dec. 30, 2022. [Online]. Available: [https://www.naspi.org/sites/default/files/2022-10/20220928\\_hariharan\\_ercot\\_NERC\\_NASPI\\_Odessa2\\_Presentation\\_FINAL.pdf](https://www.naspi.org/sites/default/files/2022-10/20220928_hariharan_ercot_NERC_NASPI_Odessa2_Presentation_FINAL.pdf)
- [14] K. Chen, N. Nayak, W. Schmus, and R. D. Quint, "Generator parameter validation (GPV)," in *NASPI Work Group Meeting and Vendor Show*, Mar. 2015, Accessed: Dec. 30, 2022. [Online]. Available: [https://www.naspi.org/sites/default/files/2016-09/02-NASPI\\_0323\\_GPV\\_031115-F.pdf](https://www.naspi.org/sites/default/files/2016-09/02-NASPI_0323_GPV_031115-F.pdf)
- [15] NASPI, "Diagnosing equipment health and mis-operations with PMU data," NASPI, technical report, May. 2015.
- [16] B. Kasztenny and I. Stevens, "Monitoring ageing CCVTs practical solutions with modern relays to avoid catastrophic failures," in *2007 Power Systems Conference: Advanced Metering, Protection, Control, Communication, and Distributed Resources*, 2007, pp. 187–208.
- [17] D. D. Shipp and T. J. Dionise, "Tutorial switching transients, transformer failures and practical solutions," Feb. 2016.
- [18] B. A. Jaradeh, "Dominion energy's pilot deployment and evaluation of enhanced linear state estimator for grid resiliency," in *EPG Webinar*, Apr. 2022, Accessed: Dec. 30, 2022. [Online]. Available: [https://www.naspi.org/sites/default/files/2022-04/D1S3\\_05\\_abu\\_20220412\\_1.pdf](https://www.naspi.org/sites/default/files/2022-04/D1S3_05_abu_20220412_1.pdf)
- [19] H. S. Saravia, A. Nath, B. A. Jaradeh, Y. Kong, Y. Lu, F. Tu, and P. Nieves, "Linear state estimator deployment for real-time power system monitoring in aep's control room," in *CIGRE UNSC's Grid of the Future Conference*, Nov. 2022, Accessed: Dec. 30, 2022.
- [20] N. Nayak, K. Martin, I. Singh, and W. Ju, "Real time applications using linear state estimation technology (RTA/LSE)," in *NASPI Work Group Meeting and Vendor Show*, Oct. 2019, Accessed: Dec. 30, 2022. [Online]. Available: [https://www.naspi.org/sites/default/files/2019-10/01\\_FOA1492\\_RealTimeApplications\\_Nayak\\_20191030.pdf](https://www.naspi.org/sites/default/files/2019-10/01_FOA1492_RealTimeApplications_Nayak_20191030.pdf)
- [21] H. Hooshyar, "Impact of synchrophasor data quality on low-frequency oscillation control," in *NASPI Work Group Virtual Meeting*, Apr. 2021, Accessed: Dec. 30, 2022. [Online]. Available: [https://www.naspi.org/sites/default/files/2021-04/D3S7\\_02\\_hosseini\\_epri\\_naspi\\_20210415.pdf](https://www.naspi.org/sites/default/files/2021-04/D3S7_02_hosseini_epri_naspi_20210415.pdf)
- [22] C. Huang, F. Li, D. Zhou, J. Guo, Z. Pan, Y. Liu, and Y. Liu, "Data quality issues for synchrophasor applications part I: a review," *J. Mod. Power Syst. Clean Energy*, vol. 4, no. 3, pp. 342–352, Jul. 2016.
- [23] N. T. Le and W. Benjapolakul, "A data imputation model in phasor measurement units based on bagged averaging of multiple linear regression," *IEEE Access*, vol. 6, pp. 39 324–39 333, Jul. 2018.
- [24] D. Osipov and J. H. Chow, "PMU missing data recovery using tensor decomposition," *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4554–4563, Nov. 2020.
- [25] B. Foggo and N. Yu, "Online PMU missing value replacement via event-participation decomposition," *IEEE Trans. Power Syst.*, vol. 37, no. 1, pp. 488–496, Jan. 2022.
- [26] P. Gao, M. Wang, S. G. Ghiocel, J. H. Chow, B. Fardanesh, and G. Stefopoulos, "Missing data recovery by exploiting low-dimensionality in power system synchrophasor measurements," *IEEE Trans. Power Syst.*, vol. 31, no. 2, pp. 1006–1013, Mar. 2016.
- [27] S. Konstantinopoulos, G. M. De Mijolla, J. H. Chow, H. Lev-Ari, and M. Wang, "Synchrophasor missing data recovery via data-driven filtering," *IEEE Trans. Smart Grid*, vol. 11, no. 5, pp. 4321–4330, Sep. 2020.
- [28] G. M. De Mijolla, S. Konstantinopoulos, P. Gao, J. H. Chow, and M. Wang, "An evaluation of algorithms for synchrophasor missing data recovery," in *2018 PSCC*, Jun. 2018, pp. 1–6.
- [29] F. Gao, J. S. Thorp, A. Pal, and S. Gao, "Dynamic state prediction based on auto-regressive (AR) model using PMU data," in *2012 IEEE Power and Energy Conference at Illinois*, Feb. 2012, pp. 1–5.
- [30] C. Huang, F. Li, L. Zhan, Y. Xu, Q. Hu, D. Zhou, and Y. Liu, "Data quality issues for synchrophasor applications part II: problem formulation and potential solutions," *J. Mod. Power Syst. Clean Energy*, vol. 4, no. 3, pp. 353–361, Jul. 2016.
- [31] K. D. Jones, A. Pal, and J. S. Thorp, "Methodology for performing synchrophasor data conditioning and validation," *IEEE Trans. Power Syst.*, vol. 30, no. 3, pp. 1121–1130, May. 2015.
- [32] M. Zaharia, R. S. Xin, P. Wendell, T. Das, M. Armbrust, A. Dave, X. Meng, J. Rosen, S. Venkataraman, M. J. Franklin, A. Ghodsi, J. Gonzalez, S. Shenker, and I. Stoica, "Apache spark: A unified engine

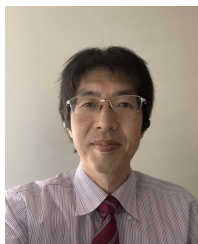
- for big data processing,” *Commun. ACM*, vol. 59, no. 11, pp. 56–65, Oct. 2016. [Online]. Available: <https://doi.org/10.1145/2934664>
- [33] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, pp. 1735–80, Dec. 1997.
- [34] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” in *3rd ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: <http://arxiv.org/abs/1409.0473>
- [35] T. Luong, H. Pham, and C. D. Manning, “Effective approaches to attention-based neural machine translation,” in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015, pp. 1412–1421. [Online]. Available: <https://aclanthology.org/D15-1166>



**YUANBIN CHENG** received a B.S. degree in Computer Science from the University of Science and Technology of China, Anhui, China, in 2016. He received his M.S. degree in Computer Science from the University of Southern California, CA, USA, in 2018. He is now pursuing his Ph.D. degree in Computer Science from the University of California, Riverside. His research interests lie in various deep learning algorithms for big data analysis in dynamic systems.



**BRANDON FOGGO (M'19)** received a B.S. degree in Electrical Engineering from the University of California, Los Angeles, in 2015, and a Ph.D. degree in Electrical and Computer Engineering from the University of California, Riverside, in 2019. His research interests lie in statistical learning theory and information theory, particularly in their merging, as well as applications to cyber physical systems with an emphasis on power distribution systems.



**KOJI YAMASHITA (M'04)** received his B.S. and M.S. degrees in Electrical Engineering from Waseda University, Tokyo, Japan, in 1993 and 1995, respectively. He also received his Ph.D. degree in Electrical and Computer Engineering from Michigan Technological University, Houghton, MI, USA, in 2020. He was a research scientist with the power system division of CRIEPI, Japan, from 1995–2018. He is currently a postdoctoral scholar at the University of California Riverside, serving as an Associate Editor of *IEEE Transactions on Power Systems*. His research interests are power system dynamic performance and its modeling with data-driven techniques.



**NANPENG YU (M'11-SM'16)** received his B.S. in Electrical Engineering from Tsinghua University, Beijing, China, in 2006. Dr. Yu also received his M.S. and Ph.D. degrees in Electrical Engineering from Iowa State University, Ames, IA, USA, in 2007 and 2010, respectively. He is currently an Associate Professor in the Department of Electrical and Computer Engineering at the University of California, Riverside, CA, USA. His current research interests include machine learning in smart grid, electricity market design and optimization, and smart energy communities. Dr. Yu is an Associate Editor of *IEEE Transactions on Smart Grid* and *IEEE Power Engineering Letters*.

• • •