Forecasting Time Series for Electricity Consumption Data Using Dynamic Weight Ensemble Model

Cheng-Hsiang Hu
Department of Computer Science and Information Engineering
National Taiwan University of Science and Technology
Taipei, Taiwan
m10615098@mail.ntust.edu.tw

Yi-Ling Chen
Department of Computer Science and Information Engineering
National Taiwan University of Science and Technology
Taipei, Taiwan
yiling@mail.ntust.edu.tw

***Abstract*—Electricity load forecasting is a prevalent research topic in recent years. In this study, we predict the electricity consumption using only previous power data (i.e., without using weather information or other features). We survey existing univariate methods such as MLP-based, CNN-based, XGBoost-based, RF-based, and EN3-bestK. However, these existing methods do not perform well due to that the range of power values varies a lot. Therefore, we present an electricity consumption forecast system called Dynamic Weight Ensemble Model (DWEM). There are three stages in the proposed DWEM. First of all, we provide three types of data serialization in data preprocessing. Second, we train four types of models (i.e., MLP-based, CNN-based, XGBoost-based, and RF-based) for building the ensemble model later. Finally, we combine the four types of models into an ensemble model, using the proposed Two-Phase Ensemble. In the two-phase ensemble, the first phase is to ensemble the models trained using the same algorithm but different serializations, and the second phase is to ensemble the models from different algorithms. The two-phase ensemble method is designed to dynamically adjust weights based on the previous performance of the corresponding models. Moreover, we notice that properly handling missing values is an important factor in system performance. Therefore, we present a statistical method to estimate the missing values. We compare DWEM with various state-of-the-art methods. Comparison of DWEM and the state-of-the-art ensemble method, the results show that DWEM is on average about 46.95% and 44.47% better than EN3-bestK on the MAPE and MAE indicators, respectively.**

***Index Terms*—Electricity Load Forecasting, Data Mining, Time Series Forecasting, Univariate**

I. Introduction

Time series forecasting research is a prevalent research topic in recent years, and one of them is electricity forecasting. Electricity forecasting can also be divided into many tasks, such as solar power output forecasting [1], and predicting the output of photovoltaic power during the daylight period. Another task is power consumption forecasting [2], which predicts the power consumption of human activities. In Taiwan, because the electricity price is very cheap [3], people tend to overuse electricity, and regional power is often tripped off in summer. In order to solve this problem, power companies try to predict the expected consumption of each household. The raw data are churning out from the smart meter. One of the motivations for our research is that electricity consumption forecasting can help residents to know what time is the peak power consumption period (and often with higher electricity price), so they can wisely schedule their power usage to lower electricity bills. Moreover, our research results can also help the government to better formulate electricity prices for different time periods based on the electricity forecasting.

Our study focuses on predictive tasks using only one variable as input, which is previous power data, and without using weather information or the other features. Univariate methods for short-term electricity consumption forecasting could be divided into two groups: the single forecasting model and the ensemble forecasting model. Example of a single forecasting model such as Sel-CNN [2], which uses a sequence of ordered power load data as the model input to forecast the next day outputs. On the other hand, ensemble forecasting model (e.g., EN3-bestK [1]) combines the predictions of the ensemble members to generate the prediction for the next day. However, the data preprocessing strategies used by EN3-bestK do not perform well in the datasets with large ranges of power load (such as Australian power load data).

For electricity load forecasting tasks, there are two problems that need to be solved. The first problem is missing power consumption values in the dataset. The existence of missing values is a common problem for time-series predictions. To solve this problem, we present a new way that uses statistical methods to estimate the missing values more properly. Our method considers three dimensions: hourly similarity, daily similarity, and weekly similarity. When comparing our results with the results generated from previous studies, our results are closer to the real electricity consumption. The experimental results show that our method is also helpful for model training and can reduce prediction errors. The second problem is the unsatisfying accuracy of prediction results on electricity consumption data. In this study, we present a novel electricity consumption forecast system called Dynamic Weight Ensemble Model (DWEM). We observe that if we train the models using different serializations on the data, and then ensemble these models to generate the forecast results, a better accuracy can be achieved. The reason is that the models trained using different serializations may have better performance in certain electrical load ranges. In addition, we

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

also observe that the models trained using different algorithms may have better performance in certain electrical load ranges. According to these observations, we leverage three types of data serialization with four different algorithms in our forecast system. In this study, we aim at the short-term electricity consumption forecasting task, which is to forecast the hourly power consumption for the next day given previous hourly power consumption data. In our forecast system, we propose a Two-Phase Ensemble approach. The first phase is to ensemble the models trained using the same algorithm but different serializations, and the second phase is to ensemble the models from different algorithms. The historical error value for each model is used to decide its weight while ensembling. The experimental results show that DWEM is better than the state-of-the-art single forecasting models and ensemble methods.

In this study, our contributions can be summarized in the following points:

- We present a new method that considers three dimensions to estimate the missing values.
- We provide three types of data serialization with four different models to increase the diversity of the ensemble model in our forecast system.
- We propose two-phase ensemble, and the experimental results show that the prediction error of the two-phase ensemble is lower than the error of the single-phase ensemble.

The organization of the paper can be summarized as follows: Related works are detailed in Section 2. In Section 3, we describe our forecast system and missing values estimation. Section 4 introduces the electricity consumption data, and also provides experiment setup and results. Conclusions are presented in Section 5.

II. Related Work

There are two research directions in electricity consumption forecasting: univariate and multivariate. The univariate approach is to use only previous power consumption data, while multivariate not only uses previous power consumption data but also uses some information from different sources. For example, weather data such as temperature and humidity or the type label of days (working days, weekends, or public holidays). In this study, we focus on the univariate approach.

Multilayer perceptron (MLP) [4] is a very classic neural network model. In [5], the authors introduce one-layered MLP with one hidden layer with 5, 5, 10 neurons to forecast three different cases, including hourly loads, total loads, and peak loads respectively.

The previous study [6] also uses one-layered ANN with 20 neurons in the hidden layer to forecast daily and hourly power consumption for 93 households in Portugal. The input data are 16 electric appliances electricity load. Their experimental results in daily energy consumption show that the average MAPE is 4.2% and the maximum MAPE is 18.1%.

CNN [7] models have outstanding performance in specific tasks, such as image recognition, speech recognition [8], and face recognition [9], but in time series forecasting tasks, few studies are proposing CNN models. The previous study [2] uses the CNN model to predict the next day of solar power data and electricity load data. They develop three different structures of CNN models to predict Australian solar data and Australian, Portuguese, and Spanish electricity data. They compared their CNN model performance with the MLP model and LSTM model. CNN model has the best performance on solar data, but on Spanish electricity data MLP model is the best.

In [10], the authors use XGBoost [11] model based on similar days to forecast power load data. Historical temperature data are used as an input to calculate the correlation of similar day and historical power data are used as the model input. Their results show that XGBoost model based on similar days can effectively predict electricity load.

Not only single models but also ensemble models are popular on electricity forecasting tasks. In [1], the authors present an ensemble method for solar power forecasting. The data was collected from the University of Queensland in Brisbane, Australia. The solar power data are sampled from 7 am to 5 pm. They aggregate the solar power data from 1-min to 30-min. Therefore, there are 20 solar power values for one day. Their approach emphasizes data diversity. They present three data preprocessing strategies includes random examples, feature sampling, and the strategy that combine the two strategies for their ensemble model members. They also present four strategies for building dynamic collections based on their ensemble model member's performance over the past 7 days. According to their experimental results, the data diversity generate by these data preprocessing strategies can effectively improve the accuracy of solar power data prediction.

For time series analysis, missing data is an inevitable problem. Therefore, how to correctly complement missing values is also a problem to be solved. The previous study [12] uses weighted moving average filter to estimate missing values. Their research is to predict the power consumption of commercial buildings. If a missing value is encountered, they use three decreasing values as the weights of the three values of the same time in the previous three days and multiply the three values by the sum after the weights as an estimate of the missing values.

III. Approach

In this section, we present the details of the proposed electricity consumption forecast system, namely, Dynamic Weight Ensemble Model (DWEM). The architecture of the DWEM is shown in Figure 1. The main inspiration for designing this electricity consumption forecast system is based on the observations that different data serializations and different machine learning models may have better performance in

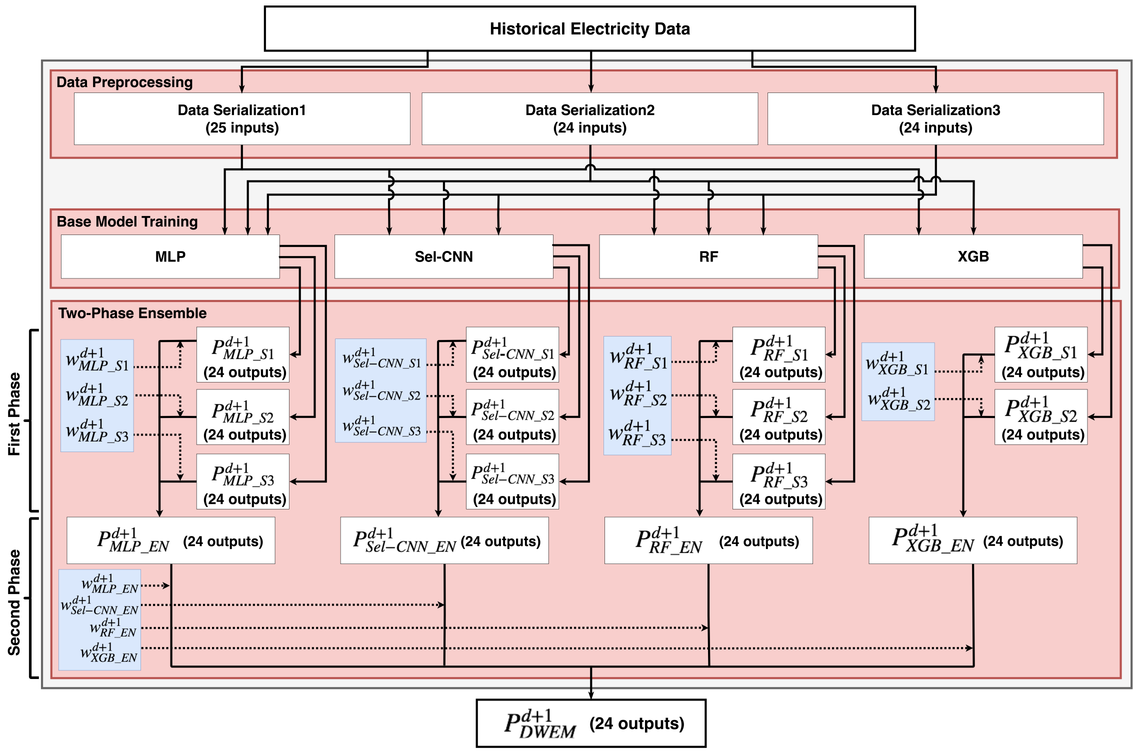


Fig. 1. DWEM system architecture.

certain electrical load areas. Some serializations and models have good performance in the peak loading area. On the other hand, some serializations and models have good performance in the normal or valley loading area. Therefore, we should give different weights to different models based on their errors with recent data in different power load intervals. As shown in Figure 1, we partition our forecast system into three stages. The first stage is data preprocessing. We convert historical electricity data into three different types of serialized vectors as input to the model. Subsequently, we use these different serialized vectors as inputs in four different models in the second stage. Finally, we use the two-phase ensemble in the third stage. The first phase is to ensemble the models trained using the same algorithm but different serializations, and the second phase is to ensemble the models from different algorithms.

A. Data Serialization

*1) **Serialization 1 (for day-to-hour prediction)**:* The procedure of Serialization 1 is as follows: We use day d and day $d-1$ hourly electricity consumption for serialization. Figure 2 shows the flowchart of Serialization 1 to predict the hourly outputs of next day. We create new sequences of vectors in which the sequence length is 25. First, we use the sequences of vectors $[p^{d-1}_{real}(t), \ldots, p^{d-1}_{real}(t+23), p^{d}_{real}(t)]$ as a model input to predict the power consumption of the first hour next day. Where $p^{d-1}_{real}(t)$ is day $d-1$ actual power consumption at time t, $p^{d-1}_{real}(t+23)$ is day $d-1$ actual power consumption at time $t+23$, and $p^{d}_{real}(t)$ is day d actual power consumption at time t. Second, we shift each value in the vector to the next hour. That is, we use $[p^{d-1}_{real}(t+1), \ldots, p^{d}_{real}(t), p^{d}_{real}(t+1)]$ to predict the power consumption of the second hour next day. The procedure goes on to complete the predictions of the 24 hourly electricity consumption values in day $d+1$.

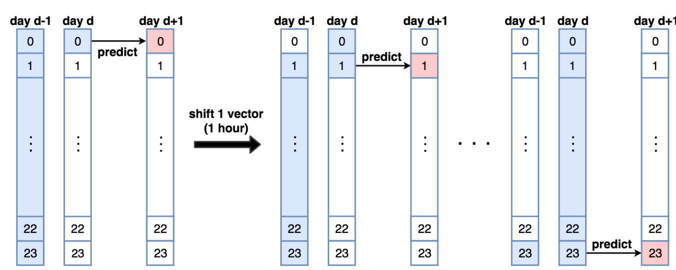


Fig. 2. Flowchart of Serialization 1

*2) **Serialization 2 (for hour-to-hour prediction)**:* The procedure of Serialization 2 is as follows: We use the power consumption of the same hour every day from day d to day $d-1$. The flowchart of Serialization 2 for predicting hourly outputs of next day is shown in Figure 3. We create new sequences of vectors in which the sequence length is 24. First, we use the sequences of vectors $[p^{d-23}_{real}(t), \ldots, p^{d-1}_{real}(t), p^{d}_{real}(t)]$ as model inputs to predict the power consumption of the first hour next day, where $p^{d-23}_{real}(t)$ is day $d-23$ actual power consumption at time t, $p^{d-1}_{real}(t)$ is day $d-1$ actual power consumption at time t, and $p^{d}_{real}(t)$ is day d actual power consumption at time

t. Second, We shift each value in the vector to the next hour. That is, we use $[p_{real}^{d-23}(t+1), \ldots, p_{real}^{d-1}(t+1), p_{real}^{d}(t+1)]$ to predict power consumption of the second hour next day (i.e., $p_{real}^{d+1}(t+1)$), and the procedure goes on to complete the predictions of the 24 hourly electricity consumption values in the day $d+1$.

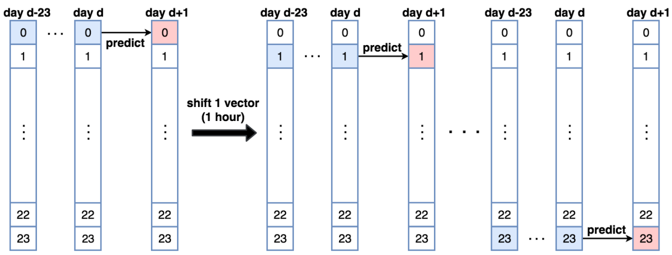


Fig. 3. Flowchart of Serialization 2

*3) **Serialization 3 (for day-to-day prediction)**:* Figure 4 shows the flowchart for Serialization 3. We use the 24 hourly power consumption values in day d as the inputs to predict the 24 hourly electricity consumption valued in day $d+1$.

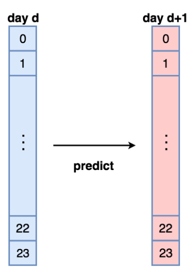


Fig. 4. Flowchart of Serialization 3

B. Base Models

We have conducted preliminary experiments to evaluate the performance of existing approaches (i.e., the ones introduced in Section II) and some popular machine learning algorithms (e.g., LSTM and linear regression). In the base model training phase, we then choose the base models with top performance as candidate base models, in order to achieve the best performance of our forecast system. Details about these four best performing base models are provided below.

*1) **MLP**:* In [5] [6], the authors present one-layered MLP to forecast power consumption in different situations. In this study, we also use MLP with one hidden layer. The input, output, and hidden layer include 24 nodes. We implement the MLP models by using Keras [13].

*2) **Sel-CNN** (Selected CNN):* In [2], the authors use Sel-CNN to forecast the next day photoVoltaic power and electricity load data. The Sel-CNN architecture is used as follows: two convolutional layers with size 24, 10, and without max pooling layers. The activation function is ReLU in the convolutional and output layers. They are using stochastic gradient descent backpropagation algorithm and the Adam optimizer in the training process. The most crucial part is that they apply batch normalization, which has a great improvement in accuracy. We use Keras [13] to implement the Sel-CNN models.

*3) **XGBoost**:* eXtreme Gradient Boosting (XGBoost) [11] is a well-known machine learning model after winning the Higgs Machine Learning Challenge. In [10], the authors use historical temperature data as an input to calculate the correlation of a similar day and use historical power data as the model input. In this study, We use Python XGBoost package to build the model. Note that, since XGBoost does not support multilabel, it can only use Serialization 1 and Serialization 2.

*4) **Random Forest**:* Random Forest (RF) [14] is a classical machine learning method for regression tasks. Random Forest creates a multitude of decision trees during training and uses the average of the predicted values of each tree as the output. In this study, we forecast the hourly power consumption of next day using only previous power data as the inputs of RF model to solve the univariate regression problem. We use RandomForestRegressor from scikit-learn (a free software machine learning library) to structure the Random Forest model.

C. Two Phase Ensemble

Figure 5 shows the weight calculation component. We calculate the error between prediction and actual power consumption in day d for each model. The weight of each model is the reciprocal of the error of each model divided by the sum of the reciprocal of the error of each model.

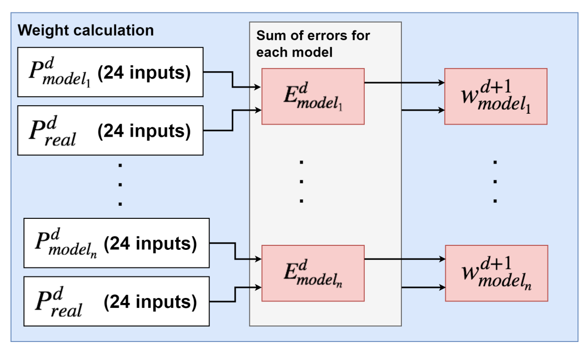


Fig. 5. The Weight Calculation Component

In the forecasting phase of our system, we use the two-phase ensemble. In the first phase, $E_{model_i}^{d}$ denotes the error of $model_i$ on day d (because $E_{model_i}^{d}$ might be zero, we add a very small value μ to avoid any division by zero), $w_{model_i}^{d+1}$ denotes the weight of $model_i$ on day $d{+}1$, $P_{model_i}^{d+1}$ denotes the prediction of $model_i$ on day $d{+}1$, and the $P_{model_i\_EN}^{d+1}$ denotes the prediction of the first phase ensemble $model_i\_EN$. The weight calculation and the ensembled result generation of the first phase ensemble are as below:

$$w_{model_i}^{d+1} = \frac{1/(E_{model_i}^{d} + \mu)}{\sum_{i=1}^{n}(1/(E_{model_i}^{d} + \mu))}$$

$$P_{model_i\_EN}^{d+1} = \sum_{i=1}^{n}(P_{model_i}^{d+1} \times w_{model_i}^{d+1})$$

For example, according to different serializations for the MLP model, there are three outputs $P_{MLP\_S1}^{d+1}$, $P_{MLP\_S2}^{d+1}$, and $P_{MLP\_S3}^{d+1}$. Then, we combine these outputs with their weights, and the MLP model first phase ensemble result is $P_{MLP\_EN}^{d+1} = P_{MLP\_S1}^{d+1} \times w_{MLP\_S1}^{d+1} + P_{MLP\_S2}^{d+1} \times w_{MLP\_S2}^{d+1} + P_{MLP\_S3}^{d+1} \times w_{MLP\_S3}^{d+1}$. Similarly, there are three outputs for the Sel-CNN model, which are $P_{Sel-CNN\_S1}^{d+1}$, $P_{Sel-CNN\_S2}^{d+1}$, and $P_{Sel-CNN\_S3}^{d+1}$. Then, we combine these outputs with their weights and the Sel-CNN model first phase ensemble result is $P_{Sel-CNN\_EN}^{d+1} = P_{Sel-CNN\_S1}^{d+1} \times w_{Sel-CNN\_S1}^{d+1} + P_{Sel-CNN\_S2}^{d+1} \times w_{Sel-CNN\_S2}^{d+1} + P_{Sel-CNN\_S3}^{d+1} \times w_{Sel-CNN\_S3}^{d+1}$.

In the second phase, the ensemble prediction of each base model in the first phase will be ensembled for the second time to generate the final prediction result. Here, $E_{model_i\_EN}^{d}$ denotes the error of $model_i\_EN$ on day d (because $E_{model_i\_EN}^{d}$ might be zero, we add a very small value μ to avoid any division by zero), $w_{model_i\_EN}^{d+1}$ denotes the weight of $model_i\_EN$ on day $d+1$, $P_{model_i\_EN}^{d+1}$ denotes the prediction of $model_i\_EN$ on day $d+1$, and P_{DWEM}^{d+1} denotes the final prediction of the DWEM. The weight calculation of the second phase ensemble and the final prediction result generation are as below:

$$w_{model_i\_EN}^{d+1} = \frac{1/(E_{model_i\_EN}^{d} + \mu)}{\sum_{i=1}^{n}(1/(E_{model_i\_EN}^{d} + \mu))}$$

$$P_{DWEM}^{d+1} = \sum_{i=1}^{n}(P_{model_i\_EN}^{d+1} \times w_{model\_EN_i}^{d+1})$$

For example, there are four different outputs $P_{MLP\_EN}^{d+1}$, $P_{Sel-CNN\_EN}^{d+1}$, $P_{RF\_EN}^{d+1}$, and $P_{XGB\_EN}^{d+1}$ from the first phase ensemble. Then, we combine these outputs with their weights, and the final prediction of DWEM is $P_{DWEM}^{d+1} = P_{MLP\_EN}^{d+1} \times w_{MLP\_EN}^{d+1} + P_{Sel-CNN\_EN}^{d+1} \times w_{Sel-CNN\_EN}^{d+1} + P_{RF\_EN}^{d+1} \times w_{RF\_EN}^{d+1} + P_{XGB\_EN}^{d+1} \times w_{XGB\_EN}^{d+1}$.

In conclusion, the first phase is to ensemble the models trained using the same algorithms but different serializations, and the second phase is to further ensemble the models from different algorithms. The advantage of the two-phase ensemble is that after the first phase, each model can predict a value that is closer to the actual power consumption value than the single model, so that the prediction results in the second phase can be better than the model that uses only single-phase ensemble. The experimental results in Table III also confirm that the two-phase ensemble method achieves higher accuracy than the model that uses the single-phase ensemble.

D. Missing Value Estimation

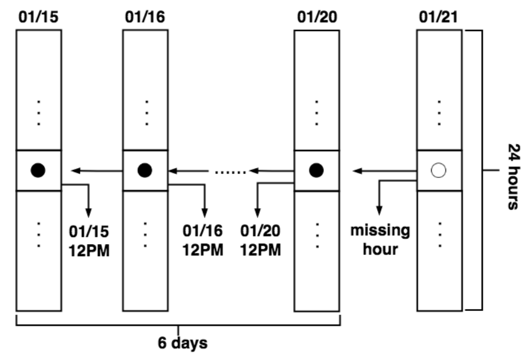


Fig. 6. Time extraction phase.

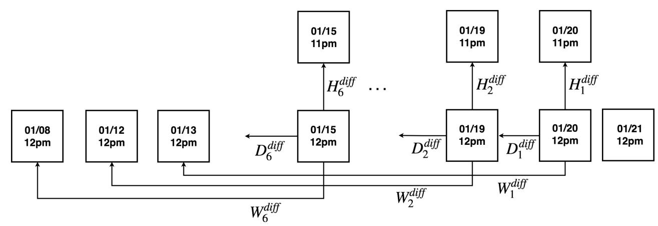


Fig. 7. Feature generation phase.

While collecting the real datasets, we notice that some datasets may have the missing value problem. For example, there are no missing values in all of the Australian datasets, but in Taiwanese dataset, there are around 9.9% missing values in each household. Additionally, the missing values in Taiwanese dataset are not similar to the missing values that we survey in other researches. The missing values in those datasets are not contiguous. They have several non-continuous hours of missing values on different days. In Taiwanese dataset, there are missing values throughout four consecutive days. Our task is to estimate the missing values. In [12], the authors use the weighted moving average filter to estimate missing values. In our case, the estimating missing value results of the weighted moving average filter is not similar to the real situation, and another problem is that different households at different times may not have the same electricity consumption behaviors. Therefore, we present a new method to estimate missing values by some statistical methods. The first phase is shown in Figure 6. In the time extraction phase, if there is a missing value, we extract the value of the same time in the previous six days. The second phase is shown in Figure 7. In the feature generation phase, we calculate three values for each value that we extract, one is the absolute value of difference from the previous hour, another is the absolute value of difference from the same time instant of the previous day, and the other is the absolute value of difference from the same time instant of the previous week. After that, for each of the above three statistics, we add to three empty lists, respectively. After daily calculations for the past six days, we calculate the standard deviation for each list H_{std}, D_{std}, and W_{std} as below:

$$
\begin{aligned}
H_{std} =& Std([H_1^{diff}, \cdots, H_6^{diff}]) \\
D_{std} =& Std([D_1^{diff}, \cdots, D_6^{diff}]) \\
W_{std} =& Std([W_1^{diff}, \cdots, W_6^{diff}])
\end{aligned}
$$

Finally, we estimate the missing values. Here, $p_{esti}^d(t)$ denotes the estimated value at time instant t on the day d, $p_{real}^d(t-1)$ denotes the value at the hour before the missing value, $p_{real}^{d-1}(t)$ denotes the value at the same time instant on the previous day, and $p_{real}^{d-7}(t)$ denotes the value at the same time instant on the same day previous week. The estimated value $p_{esti}^d(t)$ can be calculated as below:

$$
\begin{aligned}
p_{esti}^d(t) =& (p_{real}^d(t-1) \times (1/H_{std}) + \\
& p_{real}^{d-1}(t) \times (1/D_{std}) + \\
& p_{real}^{d-7}(t) \times (1/W_{std})) \\
& /(1/H_{std} + 1/D_{std} + 1/W_{std})
\end{aligned}
$$

IV. Experiment

In this section, we introduce the electricity consumption data that we use. After that, we introduce the methods used for comparison, the experiment setup, and evaluation metrics. Finally, in the results and discussions, we compare our approach with other methods using different datasets.

A. Data

Australian electricity load data we use comes from the Australian Energy Market Operator (AEMO) [15]. There are five states in the AEMO data dashboard: NSW, QLD, SA, TAS, VIC. The power consumptions are correlated with industrial, commercial, and human activities. Each state dataset we use for two years (2017 and 2018), and NSW dataset we take additional two years (2010 and 2011) which used in [2]. Each dataset is sampled every hour, and the unit of measurement is megawatt.

Taiwanese electricity load data are residential electricity load data, which are collected from Taiwan Power Company. There are 15 households electricity consumption that is sampled from January 2018 to December 2018. Each household power consumption data is sampled every minute, and the unit of measurement is watt.

B. Methods Used for Comparison

The models that used for comparison are MLP [6], Sel-CNN [2], XGBoost [10], and RF [14] and EN3-bestK [1].

The previous study [1] presents an ensemble method for solar power forecasting. The best ensemble model is EN3-bestK with linear transformation for calculating weights. EN3-bestK is constituted of 30 ensemble members, where each member is a one-layered MLP trained with different subset of data. Specifically, there are two steps in data preprocessing of EN3-bestK. They first use random sampling with replacement to sample 75% data to be a training subset. Next, for each training subset, they use random sampling with replacement, and each of the 10 ensemble members uses three different feature sampling rates of 25%, 50%, and 75% to sample features from each subset as the input of the ensemble members. For each ensemble member, they calculate and sum up the MAE over the previous seven days. They then select the K best ensemble members (K=7) based on the summation of errors, and use linear transformation for calculating the weights. The higher weights are associated with lower errors. Finally, they sum up the seven best members' predictions to be the next day solar power output. In this study, the best random example sampling rate that we use is 75%, and the best random feature sampling rate that we use is also 75%.

C. Experiment Setup

1) Australian datasets: In Australian electricity load data, we follow the previous study to split the first year (2010 & 2017) of data into training and validation set, where 70% of the data is used as training and 30% of the data is used as the validation set. We use the second year (2011 & 2018) of data for testing.

2) Taiwanese dataset: In Taiwanese electricity load data, we take the time period from January 1st to December 31st of 2018 for training. We combine 15 households' power consumption together for training and validating, in order to increase data diversity. The testing data start from July 1st to August 31st of 2019 (72 days). The reason for using July and August as testing data is that these 15 households do not have missing values in both months.

D. Evaluation Metrics

The mean absolute percentage error (MAPE) and mean absolute error (MAE) are used to evaluate the performance for comparisons, which are defined as

$$
\begin{aligned}
MAPE =& \frac{100\%}{n} \sum_{d=1}^{n} \left| \frac{y_d - \hat{y_d}}{y_d} \right| \\
MAE =& \frac{1}{n} \sum_{d=1}^{n} |y_d - \hat{y_d}|
\end{aligned}
$$

where y_d and $\hat{y_d}$ are the vectors of actual power consumption and the predicted power consumption for the day d, respectively, the number of days in the testing data is n.

E. Results and Discussion

1) Comparison of different methods for estimating missing values in training data for DWEM: The experimental results in Table II confirm that the values estimated by our method are more accurate than the values obtained by the zero filter (i.e., replacing missing values as zeros) and the weighted moving

TABLE I
RESULTS OF NSW2011, NSW2018, QLD2018, SA2018, TAS2018, VIC2018, TW2019

	NSW 2011		NSW 2018		QLD 2018		SA 2018		TAS 2018		VIC 2018		TW 2019	
Methods	MAPE	MAE	MAPE	MAE	MAPE	MAE	MAPE	MAE	MAPE	MAE	MAPE	MAE	MAPE	MAE
MLP_S3	3.44	307.01	4.80	386.03	3.39	212.29	11.15	132.75	4.38	49.77	6.58	325.12	40.30	191.63
Sel-CNN_S3	3.74	331.38	4.66	382.87	3.51	220.21	11.40	136.68	4.38	49.89	6.80	330.75	38.34	211.30
XGB_S2	3.35	306.19	3.97	330.37	2.97	192.18	10.17	130.20	4.73	53.79	5.81	296.47	41.29	187.48
RF_S2	3.44	310.97	4.02	331.29	3.05	195.09	10.61	132.90	4.62	52.37	5.95	301.17	43.14	189.13
EN3-bestK	5.15	455.05	6.56	520.83	3.64	232.30	12.06	144.88	4.99	56.26	7.44	370.29	41.80	217.55
DWEM	**2.83**	**255.01**	**3.53**	**290.75**	**2.55**	**163.13**	**9.41**	**117.43**	**4.12**	**46.76**	**5.27**	**264.54**	**32.75**	**170.41**

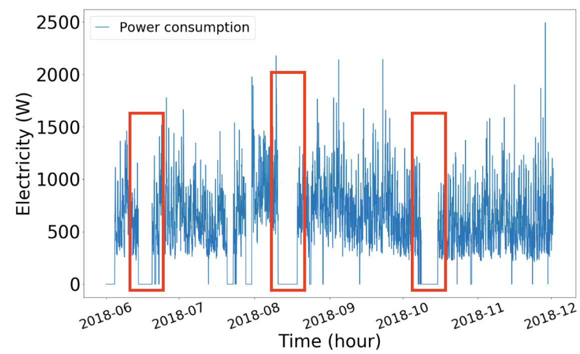
TABLE II
COMPARISON OF DIFFERENT MISSING VALUE PROCESSING METHODS IN TRAINING DATA FOR DWEM

	TW 2019	
Methods	MAPE	MAE
Zero filter	35.48	177.10
WAM filter	35.01	175.76
Our method	32.75	170.41

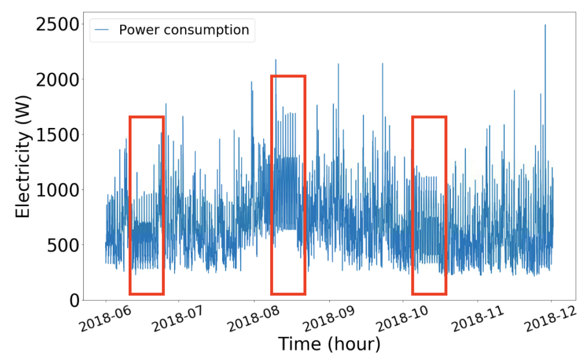
average filter. In Figure 8, we show the results of power consumption after estimated the missing values.

2) Comparison of different types of data serialization: Table IV shows the performance of the base model with different types of data serialization. For example, MLP_S1 represents MLP model with Serialization 1, MLP_S2 represents MLP model with Serialization 2, MLP_S3 represents MLP model with Serialization 3, and MLP_EN represents the output of MLP model in first phase ensemble. We use the results of NSW2011 as an example. From the table, we have two interesting observations. First of all, we find that the prediction results after the first phase, such as MLP_EN, have lower prediction errors than the prediction errors of different serializations in their models (MLP_S1, MLP_S2, and MLP_S3), and the same is true of other base models. That is to say, the results of the ensemble are better than the single model. Second, we compare different types of data serialization for each model. We can see that the best performance for MLP and Sel-CNN is Serialization 3, and the best performance for XGBoost and RF is Serialization 2. The possible reason for our discussion is that Serialization 2 uses the electricity consumption values at the same time in the previous 24 days to form a sequence as the input of the model, and these electricity consumption values are relatively similar to each other. Therefore, XGBoost and RF can make better regression predictions based on these relatively similar values. For Serialization 3, which is a sequence of 24 hours of electricity consumption values from morning to night as model input. Therefore, the neural network models MLP and CNN have better performance. According to this observation, the models used for comparison in Table I are MLP_S3, Sel-CNN_S3, XGBoost_S2, and RF_S2.

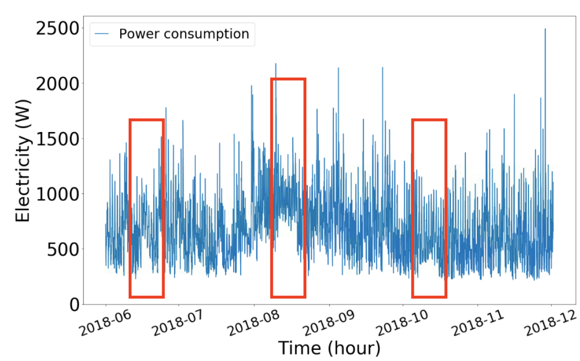
3) Comparison of various ensembles with different types of data serializations: In Table III, we compare different



(a) Missing values replaced values from zero filter



(b) Missing values replaced with estimated values from weighted moving average filter



(c) Missing values replaced with estimated values from our method

Fig. 8. Results of different methods for estimating missing values in Taiwanese dataset

TABLE III
RESULTS OF DIFFERENT MODELS WITH DIFFERENT TYPES OF DATA SERIALIZATION ENSEMBLES

	NSW 2011		NSW 2018		QLD 2018		SA 2018		TAS 2018		VIC 2018		TW 2019	
Methods	MAPE	MAE	MAPE	MAE	MAPE	MAE	MAPE	MAE	MAPE	MAE	MAPE	MAE	MAPE	MAE
DWEM_S1	4.05	357.69	4.75	385.00	3.54	223.95	11.17	136.64	4.77	54.37	6.78	337.06	40.18	191.59
DWEM_S2	3.40	304.50	4.22	350.16	3.02	194.14	10.60	135.25	4.70	52.94	5.99	303.73	36.37	177.37
DWEM_S3	3.13	280.03	4.02	326.05	3.12	195.36	10.37	124.67	4.26	46.40	6.04	297.31	36.11	177.21
DWEM_ALL	3.02	271.05	3.72	305.42	2.70	172.20	9.71	120.34	4.21	47.88	5.53	277.26	35.66	174.84
DWEM_PART	2.95	265.83	3.67	302.04	2.67	170.31	9.59	119.37	4.15	47.07	5.47	273.94	35.2	173.05
DWEM	**2.83**	**255.01**	**3.53**	**290.75**	**2.55**	**163.13**	**9.41**	**117.43**	**4.12**	**46.76**	**5.27**	**264.54**	**32.75**	**170.41**

ensemble combinations at the ensemble stage. First of all, DWEM_S1 represents the ensemble of each base model with Serialization 1. Second, DWEM_ALL represents that each base model with different serializations and a total of 11 combinations of an ensemble model. DWEM_PART represents that each base model selects the two serializations that perform best in the validation set, with a total of 8 combinations of an ensemble model. Finally, DWEM is DWEM_PART with the two-phase ensemble. The first phase is to ensemble the models trained using the same algorithm but different data serializations, and then the second phase is to ensemble the models from different algorithms. From the table, We find that when the number of base models in the ensemble model increases, the accuracy can be effectively improved, that is, DWEM_ALL is better than DWEM_S1, DWEM_S2, and DWEM_S3. However, when there are too many base models, some models result in the performance of the ensemble model worse. According to this point, we propose that each model chooses the two serializations which perform best in the validation set, that is, DWEM_PART and DWEM. Comparing DWEM with DWEM_PART, the advantage of DWEM is that after the first phase, each model can predict a value that is closer to the actual power consumption value than the single model, so that the prediction results in the second phase can be better than the model that uses the single-phase ensemble.

4) Comparison results in Australian datasets: Table I compares the performance of our ensemble methods with other state-of-the-art methods in Australian five state datasets. From the table, the results of all model predictions are proportional to the range of the power consumption data. That is, the smaller the power consumption range, the better the predicted results. The previous study [1], EN3-bestK achieves great forecasting performance due to the dataset used is solar data, which is only sampled during the daytime and sampled every half hour. The variation between the outputs produced by solar power is small, so the errors of the prediction results are not significant, which is not similar to Australian electricity load data. EN3-bestK does not perform well on the Australian dataset. The main reason is that the strategies used by EN3-bestK have feature sampling and random feature selection, which result in each member not being able to learn some range of power consumption. The larger the range of the power data, the worse the performance of EN3-bestK.

TABLE IV
RESULTS OF DIFFERENT TYPES OF DATA SERIALIZATION

	NSW 2011	
Methods	MAPE	MAE
MLP_S1	4.61	408.54
MLP_S2	3.51	313.97
MLP_S3	3.44	307.01
Sel-CNN_S1	5.03	440.57
Sel-CNN_S2	4.87	419.09
Sel-CNN_S3	3.74	331.38
XGB_S1	4.04	361.96
XGB_S2	3.35	306.19
RF_S1	4.28	381.81
RF_S2	3.44	310.97
RF_S3	3.50	315.85
MLP_EN	2.99	267.77
Sel-CNN_EN	3.51	307.64
XGB_EN	3.21	292.09
RF_EN	3.00	272.02
DWEM	**2.83**	**255.01**

Our approach DWEM is the most accurate method in Australian datasets. It is more effective in forecasting power consumption than other methods. Compared DWEM with two state-of-the-art models in the Australian datasets, the first one is Sel-CNN_S3. Our approach has average improved by 26.39% and 24.12% on the MAPE and MAE indicators. The second is EN3-bestK, and our method is 50.17% and 47.27% better than EN3-bestK.

5) Comparison results in Taiwanese dataset: Table I compares the performance of our ensemble methods with other state-of-the-art methods on an average of 15 household dataset in Taiwan. From the table, EN3-bestK is the worst. The possible reason for our discussion has as mentioned before, the data preprocessing method of EN3_BestK make some of the power load intervals and some periods of the day not selected, which resulting in EN3-bestK has the worst performance. Our method DWEM is able to combine the values predicted by different models, which achieves the lowest MAPE and MAE in Taiwanese dataset. Compared DWEM with Sel-CNN_S3 and EN3-bestK, our model has improved by 17.07% and 27.63% on the MAPE indicators, and our model has improved by 24.0% and 27.66% on the MAE indicators.

V. Conclusion

In this study, we propose Dynamic Weight Ensemble Model (DWEM), which is a novel electricity consumption forecast system. DWEM is divided into three stages. In the first and second stages of DWEM, we design three types of data serialization and use four different models. In the third stage of our system, we use the two-phase ensemble. The first phase is to ensemble the models trained using the same algorithm but different data serializations, and then the second phase is to ensemble the models from different algorithms. The strength of DWEM is to forecast the values based on adaptive weights, which are determined by the performance of the previous day for each model. The experimental results show that DWEM is on average about 46.95% and 44.47% better than EN3-bestK on the MAPE and MAE indicators, respectively.

For missing value estimation, we take household electricity consumption behavior into account. We present a new method that considers three dimensions: hourly similarity, daily similarity, and weekly similarity. The experimental results show that the estimated results from our method are closer to the actual power consumption than other methods, especially in the interval of continuous missing values.

Acknowledgment

This work was supported in part by the Ministry of Science and Technology in Taiwan through grants 109-2636-E-011-002 and 108-2221-E-011-114-MY2.

References

[1] Zheng Wang Irena Koprinska, Irena Koprinska, Alicia Troncoso, and Francisco Martínez-Álvarez. "Static and dynamic ensembles of neural networks for solar power forecasting." In 2018 International Joint Conference on Neural Networks (IJCNN), pages 1–8. IEEE, 2018.

[2] Irena Koprinska, Dengsong Wu, and Zheng Wang. "Convolutional neural networks for energy timeseries forecasting. In 2018 International Joint Conference on Neural Networks (IJCNN)," pages 1–8.IEEE, 2018.

[3] International Energy Agency(IEA) https://www.iea.org/

[4] Matt W Gardner and SR Dorling. "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences." Atmospheric environment, 32(14-15):2627–2636, 1998.

[5] Dong C Park, MA El-Sharkawi, RJ Marks, LE Atlas, and MJ Damborg. "Electric load forecasting usingan artificial neural network." IEEE transactions on Power Systems, 6(2):442–449, 1991.

[6] Filipe Rodrigues, Carlos Cardeira, and João Manuel Ferreira Calado. "The daily and hourly energyconsumption and load forecasting using artificial neural network method: a case study using a set of 93households in portugal.Energy Procedia," 62:220–229, 2014.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convo-lutional neural networks." In Advances in neural information processing systems, pages 1097–1105,2012.

[8] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. "Speech recognition with deep recurrentneural networks." In 2013 IEEE international conference on acoustics, speech and signal processing,pages 6645–6649. IEEE, 2013.

[9] Steve Lawrence, C Lee Giles, Ah Chung Tsoi, and Andrew D Back. "Face recognition: A convolutionalneural-network approach." IEEE transactions on neural networks, 8(1):98–113, 1997

[10] Xiaoqun Liao, Nanlan Cao, Ma Li, and Xiaofan Kang. "Research on short-term load forecasting usingxgboost based on similar days." In 2019 International Conference on Intelligent Transportation, BigData & Smart City (ICITBS), pages 675–678. IEEE, 201

[11] Tianqi Chen and Carlos Guestrin. "Xgboost: A scalable tree boosting system." InProceedings of the22nd acm sigkdd international conference on knowledge discovery and data mining, pages 785–794.ACM, 2016.

[12] Naveen Kumar Thokala, Aakanksha Bapna, and M Girish Chandra. "A deployable electrical load fore-casting solution for commercial buildings." In 2018 IEEE International Conference on Industrial Tech-nology (ICIT), pages 1101–1106. IEEE, 2018.

[13] Keras, https://github.com/kerasteam/keras

[14] Leo Breiman. "Random forests."Machine learning, 45(1):5–32, 2001

[15] Australian Energy Market Operator (AEMO) http://www.aemo.com.au/