

Received March 25, 2021, accepted April 5, 2021, date of publication April 8, 2021, date of current version April 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3071654

Deep Learning-Based Short-Term Load Forecasting Approach in Smart Grid With Clustering and Consumption Pattern Recognition

DABEERUDDIN SYED^{1,2}, (Member, IEEE), HAITHAM ABU-RUB², (Fellow, IEEE), ALI GHAYEB², (Fellow, IEEE), SHADY S. REFAAT², (Senior Member, IEEE), MAHDI HOUCATI³, (Member, IEEE), OTHMANE BOUHALI^{4,5}, (Member, IEEE), AND SANTIAGO BAÑALES³, (Member, IEEE)

¹Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843, USA

²Department of Electrical and Computer Engineering, Texas A&M University at Qatar, Doha 23874, Qatar

³Iberdrola Innovation Middle East, Doha 23874, Qatar

⁴Department of Research Computing, Texas A&M University at Qatar, Doha 23874, Qatar

⁵Qatar Computing Research Institute, Hamad Bin Khalifa University, Doha 23874, Qatar

Corresponding author: Dabeeruddin Syed (dsyed@tamu.edu)

This publication was made possible by the National Priorities Research Program (NPRP) Grant [NPRP10-0101-170082] from the Qatar National Research Fund (a member of Qatar Foundation), and co-funding by IBERDROLA QSTP LLC. The Open Access Funding was provided by Qatar National Library.

ABSTRACT Different aggregation levels of the electric grid's big data can be helpful to develop highly accurate deep learning models for Short-term Load Forecasting (STLF) in electrical networks. Whilst different models are proposed for STLF, they are based on small historical datasets and are not scalable to process large amounts of big data as energy consumption data grow exponentially in large electric distribution networks. This paper proposes a novel hybrid clustering-based deep learning approach for STLF at the distribution transformers' level with enhanced scalability. It investigates the gain in training time and the performance in terms of accuracy when clustering-based deep learning modeling is employed for STLF. A k-Medoid based algorithm is employed for clustering whereas the forecasting models are generated for different clusters of load profiles. The clustering of the distribution transformers is based on the similarity in energy consumption profile. This approach reduces the training time since it minimizes the number of models required for many distribution transformers. The developed deep neural network consists of six layers and employs Adam optimization using the TensorFlow framework. The STLF is a day-ahead hourly horizon forecasting. The accuracy of the proposed modeling is tested on a 1,000-transformer substation subset of the Spanish distribution electrical network data containing more than 24 million load records. The results reveal that the proposed model has superior performance when compared to the state-of-the-art STLF methodologies. The proposed approach delivers an improvement of around 44% in training time while maintaining accuracy using single-core processing as compared to non-clustering models.

INDEX TERMS Deep neural networks, distribution transformers, k-medoids clustering, machine learning, short-term load forecasting.

I. INTRODUCTION

The technological advancement in the smart grid has the goal to optimally serve the electric power generation,

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaochun Cheng.

transmission, and distribution [1]. For that, large amounts of data started to be collected from different grid sources with the intention of being utilized in various aspects such as energy forecasting [2], load analysis [3], asset management [4], customer segmentation [5], demand response management [6], energy efficiency analysis [7], anomaly

detection [8], energy trading and marketing [9], etc. The development of information technology, two-way communication system, and customer engagements will significantly increase the amount of generated and collected data of the grid [10]. The advancement of sensors has penetrated the electrical systems leading the way for smarter grids that use smart meters [11]. The massive amount of data collected by various sensors and smart meters are of high velocity, variety, veracity, value, and volume, hence satisfy all the big data characteristics [12], [13].

The rising number of installed smart meters allows for the collection of big data corresponding to consumers' end devices. The smart meter big data, representing the customer energy consumption behavior with the granularity of the household level, enable the electrical utilities to perform capacity planning, capacity building, and operations. The integration of the smart meters' capability with the communication infrastructure in smart grids enhances the protection, reliability, efficiency, and safety of the energy supply to the consumers. The collected big data have been aggregated to different levels to perform load forecasting. For aggregated feeder level forecasting, the bottom-up approach is usually implemented. In such a way, the household level consumption data are aggregated to the feeder level and then the training is performed at the feeder level. Similarly, the data at the feeder level can be aggregated to the level of the distribution transformers, while several distribution transformers could be aggregated to the level of substation and so on which helps in performing load forecasting at the needed level. The electric utilities rely on short-term forecasts at the distribution feeder and the transformer level to support peak planning and grid operation.

In this work, load forecasting is performed with the consumption data at the level of distribution transformers. The lead time of the short-term load forecasting is one day ahead, and the forecasting horizon is hourly. The hourly energy consumption prediction performed one day-ahead enables the utilities to plan and strategically structure their power system operations. Consequently, peak shaving can be structurally planned and achieved with the usage of energy storage systems and dynamic demand response units in place [14]. Load forecasting enables the electric energy utilities to plan ahead, identify the regions with high load demand, match the volatile energy demand by changing the generation capacity, reduce generation cost, regulate energy prices, and manage scheduling. Accurate load forecasting can also benefit the energy management systems to simplify control algorithms with forecasted energy signals as inputs [15].

The energy consumption may vary from one location to another owing to different weather and climate conditions. And for the same reason, the energy demand may vary on different days of the week and at different times of the day. Many researchers have been interested in grouping the different conditions or different locations based on the similarities between the available features of the data in order to reduce the number of forecasting models required for

predictions [16], [17]. The clustering techniques intrigue researchers to improve the load forecasting methodology and to enhance accuracy. The current work uses a clustering technique for a day-ahead hourly load forecasting application with the aim that the clusters and forecasting performance do not get negatively affected by the presence of outliers in the energy consumption values.

Typical methodologies employed for energy or load forecasting include time-series [18] and machine learning modeling [19]–[21]. There are two methods of machine learning modeling, namely, supervised [22] and unsupervised modeling [23]. These methodologies are worked mostly on small historical datasets. It is not clear how these methodologies can be applied to growing energy data in the era of the smart grid. The challenge is to effectively process big data from the smart grid and integrate indirect data sets including customer information, weather data, etc. into load forecasting applications. In this work, a novel method of utilizing a deep-learning supervised model along with an unsupervised machine learning technique is proposed. Firstly, to incorporate the effect of time and date on load, the past energy consumption values, termed as lag hour values, are used as features. It is followed by the pre-processing and cleaning of data. Secondly, the k-Medoids clustering technique is employed to group the transformers based on similarity in the energy consumption patterns of customers at the distribution transformers level. The clustering technique is employed to enhance the scalability of the approach. Finally, the deep learning models, including Deep Neural Networks (DNN) and Long Short-Term Memory (LSTM), are employed to train and generate predictions.

The load forecasting at the transformer level provides a pattern of the estimated load demand at the distribution network level rather than the feeder or household level. Distribution transformer load forecasting can contribute to efficient demand response management, generation scheduling, and can help in the reduction of losses. The motivation of this paper is to contribute to addressing the problem of timely and accurate short-term load forecasting in large electrical distribution networks.

To the best of our knowledge, there is no previous work that proposed a combinational hybrid methodology utilizing k-Medoids clustering algorithm and deep learning models for load forecasting. This is the first paper that proposes the use of a clustering algorithm, insensitive to the presence of outliers and solely based on energy consumption patterns, for load forecasting application in smart grids. The key contributions of this paper are summarized as follows.

- 1) A novel hybrid highly accurate forecasting approach based on clustering techniques and deep learning models is proposed. The clustering technique is aimed to enhance the scalability of the approach and its capability to analyze big data. Initially, the approach clusters the distribution transformers based on the profile of energy consumption at the aggregated level. Finally, the

forecasting models are developed within each cluster utilizing deep learning.

- 2) A pattern-based similarity metric utilizing pairwise Minkowski similarities is proposed to determine the transformers that can be clustered based on daily energy consumption patterns. This simplifies the determination of clusters that better represent the load profiles of transformers within them.
- 3) The number of clusters is optimally selected in such a way that the overall within-cluster error for all the clusters is minimized. The optimization of the sum of square errors on all transformers is performed within constraints to generate the deep learning models for each cluster.
- 4) Different from the conventional clustering method, this work avails a robust clustering algorithm insensitive to the presence of outliers.
- 5) The multi-stage methodology reduces the number of forecasting models required for predictions of energy consumption in an electric network. Hence, the methodology can be scaled to any large electric network and big data. Eventually, the proposed methodology fixes the large-scale problem, which is significant since the real-world data are usually large-scale.
- 6) We investigate the performance of the proposed scheme using real-world data and show that a gain of 44% in training time is achieved over existing schemes whilst maintaining the forecasting accuracies.

The rest of the paper is organized as follows: in Section II, the paper presents an overview of the load forecasting approaches used in the literature considering the aggregation, profiling, and clustering of the energy consumers. Section III presents the different aspects of our proposed methodology and an overview of the load forecasting approach implemented in this work while Section IV summarizes the results of the proposed clustering approach. Finally, Section V presents conclusions and future work.

II. RELATED WORK

In recent years, many researchers have invested their efforts to develop highly accurate forecasting models for energy consumption. Also, many of the presented methodologies are based on clustering using different features and conditions. In this section, the review of the proposed methodologies in the literature is presented.

Reference [17] proposed a day-ahead forecasting algorithm that uses load fluctuations and feature importance to cluster different customers at the distribution level. Crow search algorithm was utilized to determine the initialization conditions to avoid local minima convergence in the K-means clustering method and finally, an ensemble random forest model was generated to realize the day-ahead forecasting. The authors reported the lowest Mean Absolute Percentage Error (MAPE) of 1.633% for the random forest model and showed that the model performs better compared to Extreme Learning Machine (ELM), Neural Networks (NN), and

Support Vector Machines (SVM). Their methodology benefited from the clustering of the 24 hours of a day into different clusters based on the fluctuation of energy consumption. Although the employed clustering method solves the issues of criteria for selection and initialization in the k-means algorithm, there is a scope of improvement in the Crow search-based k-means clustering algorithm when faced with high multi-modal peaks in the data formulations.

In [24], the authors proposed a long-term energy forecasting methodology that utilizes the spatial clustering algorithm of Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to predict year-ahead load values for power system planning. The density-based clustering technique benefits from its inherent ability to effectively dealing with the noise in the data by eliminating the outliers. Similar sub-zones are clustered using DBSCAN based on the features of historical yearly energy consumption profiles, land use types, and geographic information. Eventually, Non-Linear Auto-Regressive (NAR) neural network models yield the values of the predicted load. They reported that their proposed model works better when compared to existing models such as exponential smoothing, grey theory, and Linear Regression (LR). However, short-term load forecasting is not addressed using this methodology.

In [25], the author proposed a hybrid model based on a Kalman filter, an artificial neural network, and wavelet transforms. The hybrid model also used clustering techniques for short-term load and renewable energy forecasting. The work provided evidence that the hybrid models involving clustering-based wavelet and artificial neural networks perform better than conventional models and other hybrid model combinations. However, in this work, the clustering was based on geographical zones, rather than the actual patterns of energy consumption.

Empirical Wavelet transformations (EWT) have been used to decompose the load data into Intrinsic Mode Functions (IMF) [26]. Along with LSTM modeling, the IMF functions are used to predict the low and medium frequency components for load predictions. Furthermore, the high-frequency components are highly varying components with uncertain characteristics, and these are clustered using Improved-DBSCAN (IDBSCAN) algorithm. The prediction results of the high, medium, and low-frequency components are aggregated to determine the total load predictions for short-term load forecasting. Their methodology has the advantage of employing different prediction methods according to the characteristics and the variance of data. However, the methodology based on IDBSCAN is not effective if the data is scaled to a large number of dimensions. Also, it is efficient only when the different clusters have varying densities.

Autoregressive Integrated Moving Average (ARIMA) model has been utilized as a baseline method for predicting energy consumption as it is easy to implement and generalize to a wide variety of specifications [27]. Nepal *et al.* used k-means clustering along with ARIMA modeling for predictions of energy consumption in buildings [27].

The clustering technique is used to cluster the days with similar load characteristics during the hours of a day. In their work, the days of a year were clustered into 6 clusters. During the prediction phase, their methodology determines the cluster number of the days preceding the testing day and finally predicts the energy consumption of the testing day. The results indicate that the standalone ARIMA model can be improved with the addition of clustering-stage as in their proposed model. However, the k-means clustering utilized is sensitive to outliers if present in the data.

Fuzzy c-Shape clustering has been investigated by Fateme *et al.* to cluster the load data depending on the shape of energy consumption [28]. A horizontal ensemble model consisting of LSTM and XGBoost has been utilized to perform a day-ahead forecast of 30-minute granular load prediction. A novel feature of apparent temperature is used in their analysis. The apparent temperature is the equivalent weather variable as experienced by humans due to the collective influence of humidity, temperature, water vapor pressure, and wind speed values. They have suggested that the addition of novel features, such as the representative feature of weather, will improve the accuracy of predictions from cluster-based ensemble models. However, their methodology is dependent on the empirical and assumed function and formulation of equivalent apparent temperature.

LSTM models have been of interest to many researchers to perform energy forecasting [29], [30]. An ensemble of LSTM was used to perform short-term energy forecasting [31]. The different branches of the ensemble utilize different clustering algorithms in their initial phases. The employed clustering algorithms involve Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH), DBSCAN, and KMeans++. In the final phase, a dense neural network is employed to aggregate the results from the different branches of the ensemble. The ensemble and deep learning models have been tested to yield better results when compared to nonensemble and classical models. In [32], Syed *et al.* proposed an averaging ensemble model of the classical algorithm including LR and deep learning algorithms including LSTM and DNNs. The results indicate that the averaging ensemble model overcomes the shortcomings of the individual models and provides synergy to enhance the overall accuracy. However, the ensemble models and the LSTMs are computationally expensive.

In [33], a novel fuzzy-based clustering method is employed to cluster data into different clusters using the order of feature importance. The clustered data goes through two different phases of regression. In the first phase, a Radial Basis Function Neural Network (RBFNN) is utilized. In the second phase, the output of the first phase is passed to a pooling layer followed by a convolutional layer and finally, through two fully connected neural networks. They tested their proposed method with two case studies to predict the hourly energy consumption for the next week with better results as compared to the classical energy forecasting methodologies. However, the clustering method utilizes common space where

data are shared between neighboring clusters and this introduces redundancy and requires additional computations.

Clustering has also been applied at the household level. In [34], Bayesian non-parametric clustering model has been applied to cluster the households with similar energy consumption profiles across seasons and neighborhoods. The load profile curves are obtained after the removal of phase variability with the application of elastic shape analysis. The household-level energy consumption has high variability and with the predictions on household-level load, it is difficult to aggregate the prediction results to high levels for the use of optimized utility operations.

In summary, there has been a significant research effort in the application of clustering techniques at different levels of energy distribution networks. The metric for clustering has been similarities in weather conditions, seasons, days of the week, hours of the day, etc. However, it is required that the metric for clustering should emphasize the patterns of energy consumption.

The main advantages of the proposed solution over similar works [35], [36] are as follows.

- 1) The clustering is solely based on the energy consumption pattern at the distribution node level in electric networks. Hence, additional granular data at the household level or other low levels of the grid are not required.
- 2) Unlike k-Means clustering, the adopted clustering is not sensitive to the presence of outliers in the data.
- 3) A trade-off between the accuracy of the predictions and the training time is achieved.

III. THE PROPOSED FORECASTING METHODOLOGY

In this section, a hybrid methodology for day-ahead hourly short-term load forecasting is proposed. Fig. 1 represents the sequence of steps performed for developing the clustering-based short-term load forecasting model. As shown in Fig. 1, the proposed methodology is carried out in four main stages:

- A. Data acquisition and pre-processing stage.
- B. Clustering stage.
- C. Training stage.
- D. Testing stage.

The time-series load data have been utilized for the case study. The methodology begins with the cleaning of data. The time-series energy consumption data consist of attributes such as entry date time and date time. The date time attribute indicates the time at which the electrical energy is consumed, and the entry date time indicates the time at which the record of energy consumption value is updated in the central distributed recording system. The irrelevant attributes, such as entry date time, meter codes, etc., in the data that do not affect energy consumption, are eliminated. The data cleaning is performed to deal with duplicate records, missing data, etc.

A. DATA ACQUISITION AND PRE-PROCESSING STAGE

The main objective of the data acquisition process is to collect the data to evaluate the proposed methodology in terms of accuracy and training time. The pre-processing stage

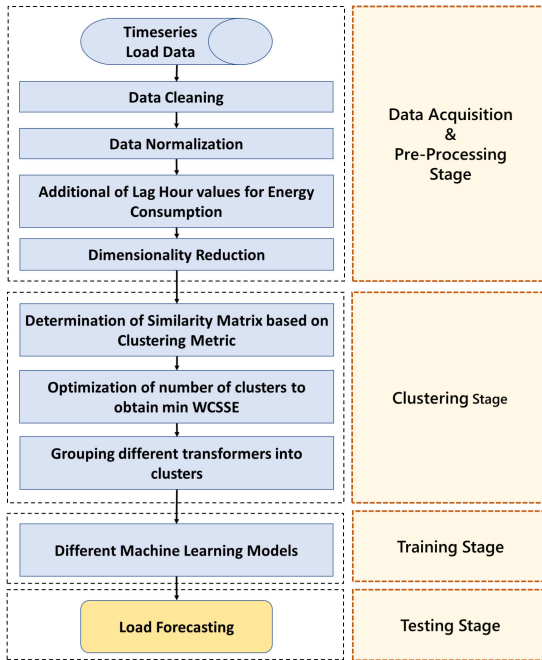


FIGURE 1. Clustering-based short-term load forecasting model.

is crucial to remove noise, avoid redundancy, and improve consistency in the data. Basically, this stage aims to enhance data quality.

The datasets used in this work, consisting of two subsets of the Spanish distribution network, have been provided by the global energy company Iberdrola. Iberdrola has been a pioneer in the deployment of Advanced Metering Infrastructure (AMI) using Power Line Communications (PLC) and open standards. The STAR project, implemented between 2008 and 2018, has mobilized an investment of 2 billion Euros resulting in 10.8 million smart meters installed and the digitization of 90,000 transformer substations [37]. Both datasets are time-series energy consumption data at the distribution transformers’ level. The major differences between the two datasets are the size of the data, and the data features. This section describes the two datasets in the following and also details the performed pre-processing steps.

1) DATA ACQUISITION

a: DATASET 1

The load forecasting data available for analysis are the energy consumption data at the distribution transformers’ level of Spain. The data contain the hourly energy consumption data for 10 distribution transformers. The weather data for the location of these 10 distribution transformers are scraped online using an Application Programming Interface (API) named Darksky [38]. The data are available for 33 months from 01 January 2017 to 28 September 2019. The weather data are merged with the energy consumption data. The data contain missing values for the weather features. The missing values should be either filled, extrapolated, or deleted [39]. The missing values for numerical features can be filled with

mean or median values. Otherwise, backward or forward filling methods can be utilized to fill the missing values. Mode imputation is applied for categorical or ordinal features. In this work, we have used the forward fill method to fill missing values for numerical weather features. Dataset 1 consists of features such as date time, wind speed, maximum temperature, minimum temperature, humidity, summertime, and other weather features in addition to energy consumption values. Fig. 2 presents the standard deviation and mean of the energy consumption values for different transformers in dataset 1. The boxes represent the deviation of the energy consumption values, the horizontal blue line inside the blue box represents the median of the consumption, and the black circles represent the outliers. The higher width of the blue boxes represents that the energy consumption values for those transformers are highly varying.

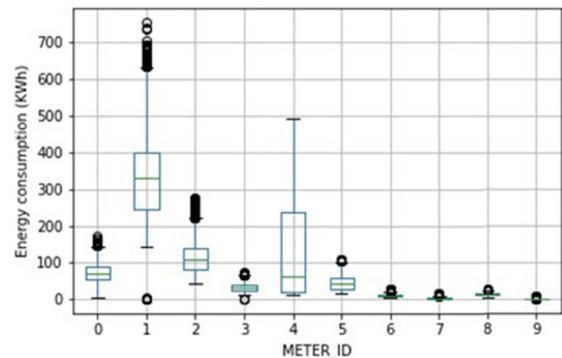


FIGURE 2. The standard deviation of energy consumption for different transformers in dataset 1.

b: DATASET 2

The load forecasting data available for analysis are the energy consumption data at the distribution transformers’ level of Spain. The data contain the hourly energy consumption data for 1000 distribution transformers. There are more than 24 million load records in the dataset. The locations of these 1000 transformers are not available currently. The data are available for the same 33 months as dataset 1. The difference for dataset 2 is that the hourly weather information at the location of 1000 transformers is not available. Hence, dataset 2 consists of features limited to energy consumption values, and season. Fig. 3 presents the standard deviation and mean of the energy consumption values for a subset of transformers in Dataset 2. The high width of the boxes in Fig. 3 depicts that the energy consumption values for transformers 21, 127, and 562 are highly varying. Moreover, it indicates that many records have zero values for consumption. The mean and the range of energy consumption values in dataset 1 and dataset 2 are mentioned in Table 1.

Fig. 4 and Fig. 5 represent the daily consumption of one transformer and the aggregated consumption of all the transformers from dataset 1 and dataset 2 respectively. It is evident that the overall pattern of consumption is the same irrespective of the consideration of one distribution transformer or

TABLE 1. Descriptive statistics of datasets.

Statistic	Dataset 1	Dataset 2
Mean (kWh)	75.25	45.41
Standard deviation (kWh)	111.87	1346.71
Minimum (kWh)	0	0
Maximum (kWh)	754.60	2147483.64

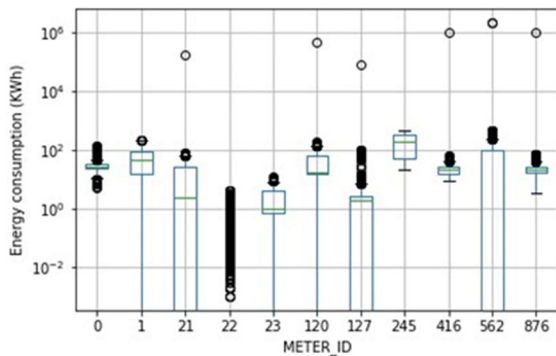


FIGURE 3. The standard deviation of energy consumption for different transformers in a sample of dataset 2.

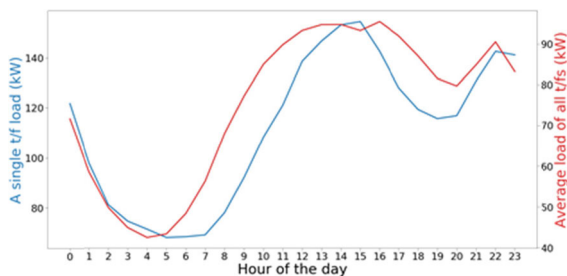


FIGURE 4. Energy consumption by the hour - dataset 1.

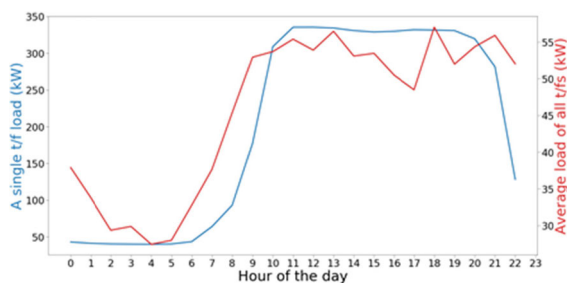


FIGURE 5. Energy consumption by the hour - dataset 2.

aggregation of the total number of transformers. As seen in Fig. 4, the energy consumption decreases gradually from the hours 00:00 to 05:00 AM and increases at a constant rate from the hours 05:00 AM to reach a peak energy consumption around 03:00 PM. The energy then declines at a constant rate till 08:00 PM and then increases with a more or less constant

slope till a peak is achieved around 10:00 PM. As per Fig. 5, the energy consumption is more or less the same between the early hours of the morning from 00:00 AM to 05:00 AM while a constant increase in the energy demand is witnessed until 10:00 AM. The energy consumption then fluctuates between a small range between 10:00 AM and 09:00 PM with local peaks observed at 01:00 PM and 06:00 PM. After 09:00 PM, there is a decline in the consumption of energy.

2) DATA PRE-PROCESSING

It involves the stages of data cleaning, scaling attributes, attribute/feature selection, feature extraction, etc.

a: DATA NORMALIZATION

For accurate and efficient learning of machine learning algorithms, it is required that all the attributes have the same numerical contribution and variance in the same order. If one attribute has variance much larger than another attribute, then it dominates whilst learning the objective function. To incorporate a non-distorting scaler, Minimum-Maximum (min-max) Scaler has been utilized in this work. The min-max scaler is given as per (1).

$$a'_m = c + \frac{(a_m - \min(a))(q - p)}{(\max(a) - \min(a))} \tag{1}$$

where a'_m is the new attribute value at row m , a_m is the original attribute value at row m , $\min(a)$ is the minimum value of the attribute, $\max(a)$ is the maximum value of the attribute, and $[p, q]$ is the scaling range decided for the attribute a .

b: ADDITION OF LAG FEATURES

Lag hour energy consumption values are added in the pre-processing step to provide additional feedback features to the model. The optimal number of lag hour values is determined to be in the order of 24.

c: FEATURE SELECTION

The candidate input features in dataset 1 include the lag energy consumption values, wind speed, maximum temperature, minimum temperature, instantaneous temperature, humidity, pressure, wind speed, wind gust, cloud cover, precipitation intensity, dewpoint, Ultraviolet (UV) index, visibility, season, and calendar variables. There are 39 input features in the candidate set of inputs.

The candidate input features in dataset 2 include the lag energy consumption values, season, and calendar variables. There are 28 input features in the candidate set of inputs.

For feature selection, two methods have been investigated. The feature importance scores are calculated utilizing the permutation feature importance techniques. Additionally, a top-down search based feature selection method called Sequential Backward Search (SBS) is employed to address the multi-collinearity between the different features unlike the best individual feature technique [32].

B. CLUSTERING STAGE

The pre-processed data with high quality in terms of consistency, low noise, and no redundancy are passed to the next stage of clustering. The main objective of this stage is to group together the different distribution transformers with similar patterns in the daily energy consumption. The similarity metric utilized to group the transformers together is the daily energy consumption. There are two factors to be kept in mind while clustering-technique is used - the number of clustered models and accuracy. As observed in dataset 2, there are 1000 transformers. If each of the transformers is modeled separately, then these models capture the patterns of energy consumption for the transformers effectively. However, the aim is to reduce the number of models to be developed from 1000 to as much low number as possible. Nevertheless, while reducing the number of clustered models to be created, it is also crucial to maintain the accuracy of the models. Two of the clustering algorithms have been considered in the time-series forecasting of energy consumption at the transformers level. These algorithms are described in the following:

Algorithm 1 . K-Means Algorithm

Input: $\epsilon, k, \text{Data } S$

Initialize k centroids randomly, $\mu_1^t, \mu_2^t, \dots, \mu_k^t \in \mathbb{R}^d$

Output:

```

1: while  $\sum_{i=1}^k \left| \mu_i^t - \mu_i^{t-1} \right|^2 \leq \epsilon$  do
2:    $t \leftarrow t + 1$ 
3:    $C_j \leftarrow \Phi$  for all  $j = 1, 2, 3, \dots, k$ 
4:   for all  $x_p \in S$  do
5:      $p^* \leftarrow \operatorname{argmin}_p \{ |x_p - \mu_i|^2 \}$ 
6:      $C_{p^*} \leftarrow C_{p^*} \cup x_p$ 
7:   end for
8:   for all  $i = 1 \text{ to } k$  do
9:      $\mu_i^t \leftarrow \frac{1}{C_i} \sum_{x_p \in C_i} x_p$ 
10:  end for
11: end while

```

1) K-MEANS CLUSTERING ALGORITHM

The objective of the k-Means clustering algorithm [40] is to reduce the Error Sum of the Squares (SSE) scoring function that is given by (2).

$$SSE = \sum_{i=1}^k \sum_{x_p \in C_i} \|x_p - \mu_i\|^2 \quad (2)$$

where k represents the total number of clusters, C_i represents each cluster, x_p represents each point in a cluster, and μ_i is mean of all points in a cluster.

K-Means applies an iterative greedy approach to reduce the sum of squares error until it reaches a local optimum. k-Means starts with the selection of the number of clusters k and the initial k number of centroids assigned to each cluster. This step is followed by the centroid update. At this stage, all the points are assigned to the clusters with the nearest centroids. Once all the points are assigned, the centroids

are updated for each cluster as the mean values of all the points in the clusters. The cluster assignment and the centroid update are repeated until there is no change in the centroids in two subsequent loops. This indicates the point of local minima.

The algorithm for the k-Means model is given in Algorithm 1. The value of k is selected in such a way that the average distance from points to centroid decreases rapidly till it converges or changes slowly thereafter.

2) K-MEDOIDS CLUSTERING

It is known that means, as a statistic, is highly sensitive to the outliers. The k-Means algorithm, that determines and utilizes the means of the data points in calculations, is particularly sensitive to the outliers in the data. To overcome this, a technique of using medoids instead of average values in a cluster is devised. Medoids are centrally located points in a cluster and the technique is called the k-Medoids clustering. Although k-Medoids are computationally more demanding, k-Medoids clusters are not particularly sensitive to the presence of outlier points and are applicable to both continuous and discrete domains of data [41]. This algorithm minimizes the sum of dissimilarities between the objects in a cluster with the reference object selected for that cluster. Basically, the input given is the value of k that represents the number of clusters defined for the data. For each of the k clusters, k -reference points are selected. The remaining points are clustered into the cluster of a reference point such that the sum of the dissimilarities between the reference object and the points in the cluster is minimized. With different initial medoids selected, the clusters obtained are different. The difference between the k-Means algorithm and the k-Medoids algorithm is that k-Means consider the average value in a cluster to be a reference point and k-Medoids consider the points to be a reference object for the clusters. Algorithm 2 presents the sequence of steps performed in the k-Medoids Algorithm.

Algorithm 2 . K-Medoids Algorithm

Input: $\epsilon, k, \text{Data } S$

Initialize k medoids randomly, $\Theta_1^t, \Theta_2^t, \dots, \Theta_k^t \in \mathbb{R}^d$

Output:

```

1: while  $\sum_{i=1}^k \left| \Theta_i^t - \Theta_i^{t-1} \right|^2 \leq \epsilon$  do
2:    $t \leftarrow t + 1$ 
3:    $C_j \leftarrow \Phi$  for all  $j = 1, 2, 3, \dots, k$ 
4:   for all  $x_p \in S$  do
5:      $p^* \leftarrow \operatorname{argmin}_p \{ |x_p - \Theta_i|^2 \}$ 
6:      $C_{p^*} \leftarrow C_{p^*} \cup x_p$ 
7:   end for
8:   for all  $i = 1 \text{ to } k$  do
9:      $\Theta_i^t \leftarrow \frac{1}{C_i} \sum_{x_p \in C_i} x_p$ 
10:  end for
11: end while

```

C. TRAINING STAGE

The main objective of this stage is to develop machine learning and deep learning networks and train these networks on real training data. After the similar transformers are clustered together, the pre-processed data are passed to linear regression, deep neural networks, and long short-term memory networks to train the models for STLF. These models are explained in the following:

1) LINEAR REGRESSION (LR)

The hypothesis of LR is given by:

$$h_w(x) = w_0 + w_1x_1 + w_2x_2 + \dots + w_nx_n \quad (3)$$

where x_1, x_2, \dots, x_n represent features, and $w_0, w_1, w_2, \dots, w_n$ represent model parameters.

The objective of the LR is to choose $w_0, w_1, w_2, \dots, w_n$ so that the values of $h_w(x)$ is as close to the actual values of the labels (y). This is achieved by the introduction of a constraint while determining $w_0, w_1, w_2, \dots, w_n$.

The objective constraint is given by (4) [42].

$$J(w_0, w_1, \dots, w_n) = \frac{\text{minimize } 1}{w_0w_1 \dots w_n 2n} \times \sum_{k=1}^n (h_w(x^{(k)}) - y^{(k)})^2 \quad (4)$$

Here, $J(w_0, w_1, \dots, w_n)$ is the cost function in terms of model parameters.

This constraint is basically the sum of squared error and the aim is to minimize this error while determining the weights.

The LR has been used as one of the prediction models to act as a benchmark for training time as this model would have the lowest training time owing to the simplicity of the model but coarser accuracy.

2) DEEP NEURAL NETWORKS

If the artificial neural networks have multiple hidden layers between the input layer and the output layer, then these are termed as Deep Neural Networks (DNNs) [43]. DNNs have the capabilities of modeling linear and non-linear relationships between the data features. Further, the tendency to overfit can be reduced with the application of dropout where the neurons are dropped in random or systematic order [44].

The non-linear function representing the data is effectively determined in the neural networks using summation and product operations. If a neuron 'j' of layer 'l' (depicted in Fig. 6) from a neural network is considered, then the input to the neuron is S_j^l , the weight at the neuron is w_{ij}^l . Let σ be the activation function, then x_j^l is the output from the neuron and this output acts as input to the neurons in the next layer. Here, i represents the neuron number in the previous layer and d^l represents the number of neurons in the layer 'l'.

The input to the neuron S_j^l is given as

$$S_j^l = \sum_{i=1}^{d^{(l-1)}} w_{ij}^{(l)} x_i^{(l-1)} + b_j^{(l)} = \sum_{i=0}^{d^{(l-1)}} w_{ij}^{(l)} x_i^{(l-1)} \quad (5)$$

and the output from the neuron j in layer l is given as

$$x_j^l = \sigma(S_j^l) \quad (6)$$

In the matrix form, the equation for the input to neuron in layer l is given as:

$$\underline{S}^l = (W^l)^T \underline{x}^{l-1} \quad (7)$$

This equation is used in the forward propagation calculations. The value \underline{x} , which is input, is available initially. It is used with pre-initialized weights $W^{(1)}$ to calculate the input $\underline{S}^{(1)}$ to the neurons in the hidden layer 1. This input when applied with activation function yields the output $\underline{x}^{(1)}$ from the neuron in hidden layer 1.

Algorithm 3 . Forward Propagation

Input: $\underline{x}^{(0)} = \underline{x}$. Initialization of augmented vector

Output:

- 1: **for** $l = 1, 2, 3, \dots, L$ **do**
 - 2: compute $\underline{S}^l = (W^l)^T \underline{x}^{l-1}$
 - 3: compute $\underline{x}^{(l)} = \begin{bmatrix} 1 \\ \sigma(\underline{S}^l) \end{bmatrix}$
 - 4: **end for**
 - 5: compute $h(\underline{x}) = \underline{x}^{(L)}$
-

Graphically, the forward propagation can be represented as the calculations involving the following:

$$\underline{x} \equiv \underline{x}^{(0)} \xrightarrow{W^{(1)}} \underline{S}^{(1)} \xrightarrow{\sigma} \underline{x}^{(1)} \xrightarrow{W^{(2)}} \underline{S}^{(2)} \xrightarrow{\sigma} \underline{x}^{(2)} \dots \xrightarrow{W^{(L)}} \underline{S}^{(L)} \times \xrightarrow{\sigma} \underline{x}^{(L)} \equiv h(\underline{x}) \quad (8)$$

The algorithm for the forward propagation of the neural network is given in Algorithm 3. The aim of the forward propagation is to calculate the inputs and outputs in different layers of the network using the weights, bias, and activation functions.

Algorithm 4 . Neural Network

Input: Initialize all weights $w_{ij}^{(l)}$

Output:

- 1: **for** $t = 0, 1, 2, 3, \dots$, **do**
 - 2: pick $n \in 1, 2, 3, \dots, N$
 - 3: Forward propagation: compute all x_j^l
 - 4: Backward propagation: compute all δ_j^l
 - 5: update the weight: $w_{ij}^{(l)} \leftarrow w_{ij}^{(l)} \eta x_i^{l-1} \delta_j^l$
 - 6: iterate to the next step until it is time to stop
 - 7: **end for**
 - 8: return the final weights $w_{ij}^{(l)}$
-

The backpropagation is utilized to determine the gradient of error in the direction from the last hidden layer to the first hidden layer while minimizing the gradient of the error with respect to the weights of neurons.

The error associated with the predictions is given by (9) [45]. The subsequent equations (10), (11), and (12) are

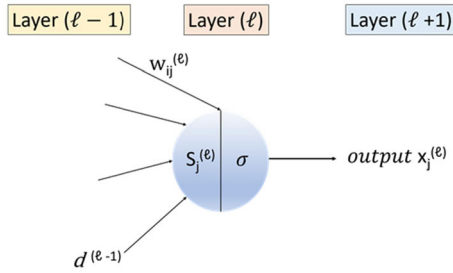


FIGURE 6. Neuron j of layer l in a neural network.

the calculation of error gradient with respect to the weight of the corresponding layer.

$$E = \frac{1}{N} \sum_{n=1}^N \frac{\partial e_n}{\partial W} (h(x_n - y_n)^2) \quad (9)$$

$$\frac{\partial E}{\partial W^l} = \frac{1}{N} \sum_{n=1}^N \frac{\partial e_n}{\partial W^l} \quad (10)$$

where $\frac{\partial e_n}{\partial W^l}$ is given by $\frac{\partial e_n}{\partial w_{ij}^l}$ that is equal to

$$\frac{\partial e_n}{\partial w_{ij}^l} = \frac{\partial e_n}{\partial s_j^l} \frac{\partial s_j^l}{\partial w_{ij}^l} \quad (11)$$

$$\frac{\partial e_n}{\partial w_{ij}^l} = \frac{\partial e_n}{\partial s_j^l} x_i^{l-1} \quad (12)$$

This brings the partial derivative of the error with respect to neuron weights to the following equation:

$$\frac{\partial e_n}{\partial w_{ij}^l} \equiv \frac{\partial e}{\partial W^l} = \delta_i^l x_i^{l-1} \quad (13)$$

In the backpropagation, the error gradient $\delta_i^{(L)}$ is determined first (L represents the last layer in the neural network) and by way of backpropagation, the errors in the previous layers are calculated as the following:

$$\begin{aligned} \delta_i^{l-1} &= \frac{\partial e(w)}{\partial s_i^{l-1}} \\ &= \sum_{j=1}^{d^l} \frac{\partial e(w)}{\partial s_j^l} \frac{\partial s_j^l}{\partial x_i^{l-1}} \frac{\partial x_j^l}{\partial s_i^{l-1}} \\ &= \sum_{j=1}^{d^l} \delta_j^l w_{ij}^l \frac{\partial x_j^l}{\partial s_i^{l-1}} \end{aligned} \quad (14)$$

The above equation is the representation of the error gradient of a layer in terms of the error gradient of the next layer. All the steps of forming a DNN are provided in Algorithm 4.

3) LONG SHORT TERM MEMORY (LSTM) NETWORKS

LSTM is a type of Recurrent Neural Network (RNN) that predicts the output based on not only the current state of the hidden units but also on the previous states witnessed so far, with the help of storing information in memory blocks [46].

LSTMs are sequential models and hence, capture the temporal dependencies. These models are suitable for processing time-series data such as load forecasting data. In a standard RNN, there are two inputs at a time step t to a neuron: input of time step t (x_t) and output obtained at time step $t - 1$ (h_{t-1}). Output at a time step is obtained by the weighted sum of x_t and h_{t-1} which is then followed by using activation functions such as Rectified Linear Unit (ReLU), hyperbolic tangent (\tanh), etc. on the weighted sum.

LSTM places a mini neural network inside each neuron and therefore complicates the process of training. However, it helps to improve reliance and handles the long-term dependencies well by eliminating the issues of gradient vanishing and gradient explosion that usually exist with the use of standard RNN. The main idea of LSTM is to have two outputs and gates. One of the outputs goes to the output layer and the next time step. Besides, the other output goes to the next time step only. Gates are the multiply operations performed and there are several gates in the LSTM. The LSTM network determines the weights and these weights are used to dot-product the inputs.

An LSTM layer followed by a fully connected neural network is depicted in Fig. 7. The machine learning models utilized in the proposed clustering-based modeling and their parameters are specified in Table 2. The values for the parameters are obtained after grid search parameter optimization. The objective of the work is to develop day-ahead hourly forecasting models whilst minimizing mean squared error as the loss function.

TABLE 2. Machine learning models used in clustering-based short-term load forecasting (STLF).

Model Name	Specifications	Type
Deep Neural Network	Total layers: 6 No. of neurons in hidden layers: {75, 50, 30, 20}	DNN
LSTM	Total layers: 6 No. of neurons in hidden layers: {52, 42, 42, 32}	RNN
Linear Regression		Classical Model

D. TESTING STAGE

In this stage, the performance of the clustering-based deep learning models is evaluated by testing these models on the datasets of all the distribution transformers. The performance of clustering-based models is compared against the individual models developed for each transformer. The metrics of evaluation utilized for accuracy are Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE). Training time and testing time are used to evaluate the performance in terms of execution time.

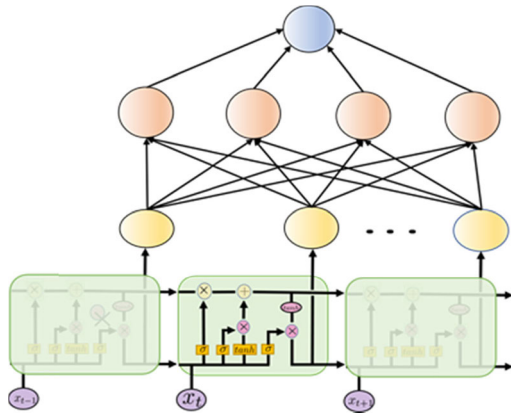


FIGURE 7. LSTM with fully connected NN.

- 1) Root Mean Square Error (RMSE): RMSE is the square root of the sum of squares of the difference between actual and predicted energy consumption. RMSE is an effective performance metric for comparing forecasting errors of different models for a single attribute which is the case in our paper. However, it is not a recommended measure to compare performance between attributes as RMSE is scale dependent. RMSE is given by (15) [47].

$$RMSE = \sqrt{\frac{1}{N} \sum_{r=1}^N (E_{predicted} - E_{actual})^2} \quad (15)$$

- 2) Mean Absolute Percentage Error (MAPE): MAPE represents the ratio (in percentage) of the absolute difference between the actual and predicted value to the actual value at every record of energy consumption. It is necessary to make sure that the actual value is not zero while calculating MAPE. MAPE is given by (16) [47].

$$MAPE (\%) = \frac{100}{N} \sum_{r=1}^N \left| \frac{E_{predicted} - E_{actual}}{E_{actual}} \right| \quad (16)$$

Here, E_{actual} is the actual energy consumption, $E_{predicted}$ is the predicted value of energy consumption, and N is the number of energy values. For low prediction values, the MAPE value cannot exceed 100%. However, for high prediction values, there is no maximum control limit to the value of MAPE.

The data of each transformer are split into a training set and a testing set with the data split as 80% and 20% respectively. Algorithm 5 details the sequence of steps designed in the proposed methodology to perform the STLF task.

IV. RESULTS OF CLUSTERING-BASED MODELS

The current work focuses on the application of clustering by energy consumption patterns at the transformer level to enhance the performance of forecasting. The methodology determines the clusters of similar transformers based on a similarity metric of aggregate daily energy consumption after which the data are sent to the machine learning models. The proposed methodology is applied to two real datasets

to illustrate the performance and enhancement in the forecasting accuracy. The computations are performed on 1 core with 32 GB RAM.

A. SIMILARITY METRIC FOR TRANSFORMERS

At the outset of the clustering stage, a similarity matrix is required to define the similarities and dissimilarities between the transformers. The similarity metric is defined to be aggregated daily load to capture the daily load patterns of the transformers. Let $L_{s,t}$ be the load of transformer s at time duration t . The load matrix for the transformers is represented by $L_{S \times T}$ in (17) where S is the total number of transformers and T is the total number of days when aggregation level is 24 hours. The size of the matrix increases with the increase in the number of transformers or with the reduction in the aggregation level of energy consumption values. For the 1000 transformers dataset that is used in case study 2, the size of the load matrix is 1000×1001 where 1001 is the number of days between 01 January 2017 to 28 September 2019.

$$L_{S \times T} = \begin{bmatrix} L_{1,1} & L_{1,2} & \dots & L_{1,t-1} & L_{1,t} \\ L_{2,1} & L_{2,2} & \dots & L_{2,t-1} & L_{2,t} \\ \vdots & \vdots & \dots & \vdots & \vdots \\ L_{S-1,1} & L_{S-1,2} & \dots & L_{S-1,t-1} & L_{S-1,t} \\ L_{s,1} & L_{s,2} & \dots & L_{s,t-1} & L_{s,t} \end{bmatrix} \quad (17)$$

The similarity between any two transformers r, s at any given time p is determined based on pairwise Minkowski similarity which is given by (18).

$$D_{r,s,p} = \left(\sum_{t=1}^{24} |L_{r,t} - L_{s,t}|^q \right)^{\frac{1}{q}} \quad (18)$$

where L_r, L_s represent the row vectors of load values for transformers r, s respectively. The optimized value of q in (18) was determined to be equal to unity. Finally, the obtained pairwise similarity matrix is passed as an argument to the clustering function to obtain the clusters of transformers with similar energy consumption patterns. The adoption of Minkowski similarity enhanced the performance of clustering.

B. OPTIMIZATION OF THE NUMBER OF CLUSTERS

There are various methods to optimize the number of clusters (k) in a clustering algorithm. There are direct methods such as elbow curve [48], average silhouette [49], etc., and statistical methods such as gap statistic [50], etc. The direct methods involve optimizing a cost function such as the minimization of within-cluster error. Statistical methods are those methods that collect evidence to support a hypothesis or to reject a null hypothesis [51]. In this work, the elbow curve is constructed to determine the optimal number of clusters and the results are shown in the following:

1) ELBOW CURVE

In this work, the direct method of the elbow curve is utilized. The elbow curve calculates the within-cluster sum of

Algorithm 5. Proposed Methodology**Input:**

D: hourly energy consumption data for an electrical network.

S: total number of transformers.

Output:

```

1:  $D^{averaged} \leftarrow \text{rowByRowWeightedAverage}(D)$ ;
2:  $L \leftarrow \text{concatenate}(D^{averaged})$ ;
3:  $\text{SimMat} \leftarrow \text{minkowskiDistances}(L, L)$ ;
4: while  $k < S/2$  do
5:    $k_{\text{opt}} \leftarrow \text{argmin}_k ||\text{calculateWCSSE}(\text{SimMat}, k)||$ ;
6: end while
7:  $C \leftarrow k\text{Medoids}(k_{\text{opt}}, \text{SimMat}).\text{get\_clusters}()$ ;
8: for  $\text{iter1} = 0$  to  $\text{length}(C)$  do
9:    $\text{globals}()[D_{\text{train}}^{\text{iter1}}] \leftarrow \text{null}$ ;
10:   $\text{globals}()[D_{\text{test}}^{\text{iter1}}] \leftarrow \text{null}$ ;
11:  for  $\text{iter2} = 0$  to  $\text{length}(C[\text{iter1}])$  do
12:     $\text{tfnum} \leftarrow C[\text{iter1}][\text{iter2}]$ ;
13:     $D^{C[\text{iter1}][\text{iter2}]} \subset D \forall \text{tfnum}$ ;
14:     $\text{globals}()[D_{\text{train}}^{\text{tfnum}}] \leftarrow 80\% \text{ of } D^{C[\text{iter1}][\text{iter2}]}$ ;
15:     $\text{globals}()[D_{\text{test}}^{\text{tfnum}}] \leftarrow 20\% \text{ of } D^{C[\text{iter1}][\text{iter2}]}$ ;
16:    Train machine learning model on train data  $D_{\text{train}}^{\text{tfnum}}$ ;
17:    Store tuned hyperparameters for further use.
18:    Dump trained model  $M_{\text{individual}}^{\text{tfnum}}$  for future testing.
19:    Forecast on test data  $D_{\text{test}}^{\text{tfnum}}$  with  $M_{\text{individual}}^{\text{tfnum}}$ ;
20:     $\text{globals}()[D_{\text{train}}^{\text{iter1}}].\text{append}(\text{globals}()[D_{\text{train}}^{\text{tfnum}}])$ ;
21:     $\text{globals}()[D_{\text{test}}^{\text{iter1}}].\text{append}(\text{globals}()[D_{\text{test}}^{\text{tfnum}}])$ ;
22:  end for
23:  Train ML model on cluster train data  $\text{globals}()[D_{\text{train}}^{\text{iter1}}]$ .
24:  Store tuned hyperparameters for further use.
25:  Dump trained model  $M_{\text{clustered}}^{\text{iter1}}$  for future testing.
26: end for
27: for  $\text{iter1} = 0$  to  $\text{length}(C)$  do
28:  for  $\text{iter2} = 0$  to  $\text{length}(C[\text{iter1}])$  do
29:     $\text{tfnum} \leftarrow C[\text{iter1}][\text{iter2}]$ ;
30:    Forecast on test data  $\text{globals}()[D_{\text{test}}^{\text{tfnum}}]$  with  $M_{\text{clustered}}^{\text{iter1}}$ ;
31:    Compare the accuracy results of  $M_{\text{clustered}}^{\text{iter1}}$  and  $M_{\text{individual}}^{\text{tfnum}}$  for  $\text{tfnum}$ ;
32:  end for
33: end for

```

square errors (WCSSE) and determines the k -value such that WCSSE is minimized. The aim of the selection of k is to determine a low value of k such that the sum of square error for that value of k is the minimum and if any more clusters are added, the clustering is not improved much. This is to provide a trade-off between the number of clusters and the accuracy. The elbow method is selected over other methods of determining the k -value for clustering because of its low complexity. As per existing research [52], the execution time is the lowest for the elbow method when compared to other methods owing to its low complexity of utilizing the sum of

the square distance between cluster points and representative centers.

To determine the optimal number of clusters, the elbow curves are obtained for dataset 1 and dataset 2 as illustrated in Fig. 8 and Fig. 9. The independent axes in the figures indicate the number of clusters and the dependent axes in the figures represent the WCSSE for the corresponding number of clusters (k) value. As per Fig. 8, the sharp decline in the WCSSE is observed for $k = 3$. Hence, the optimum number of clusters is selected as 3 for 10 transformers dataset. The elbow in Fig. 9 suggests that the optimal number of clusters is $k = 93$ for the 1000 transformers dataset. Hence, the clusters are determined, and the deep learning models are developed with the number of clusters $k = 93$.

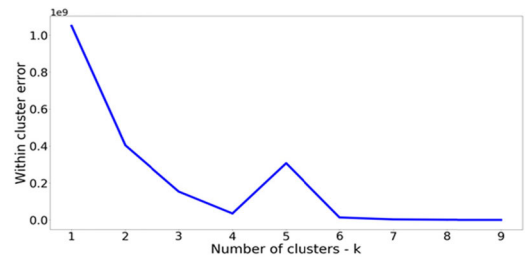


FIGURE 8. Elbow curve for 10 transformers dataset.

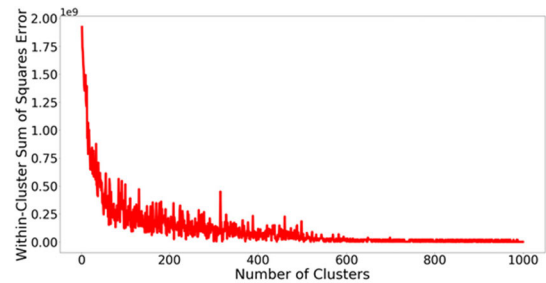


FIGURE 9. Elbow curve for 1000 transformers dataset.

C. EVALUATION OF FEATURE ENGINEERING

Fig. 10 depicts the feature importance scores determined for the weather attributes and this illustrates that temperature, pressure, humidity, and UV index are the most crucial attributes contributing to the accurate predictions of the target variable whilst learning the objective function. Fig. 11 illustrates the feature importance scores determined for the lag consumption attributes for 6-hour ahead prediction of energy consumption values. The accuracies of forecasting models increase after the application of feature engineering i.e. feature selection using Sequential Backward Search (SBS). Fig. 12 illustrates the RMSE performance of forecasting models for increasing number of features in the data. The forecasting models are developed after the application of the proposed clustering-based deep learning approach. However, the results are shown after testing on the individual transformers within the clusters. The best testing

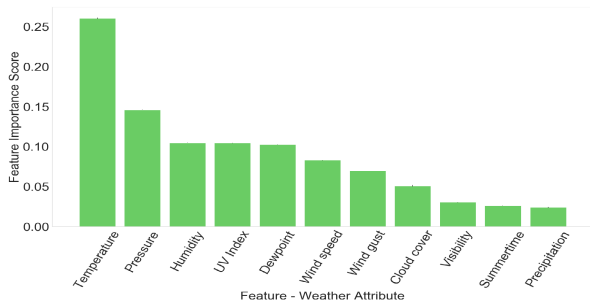


FIGURE 10. Feature importance - weather attributes.

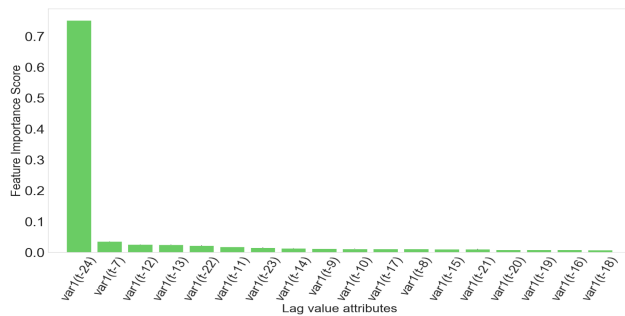


FIGURE 11. Feature importance - lag value attributes.

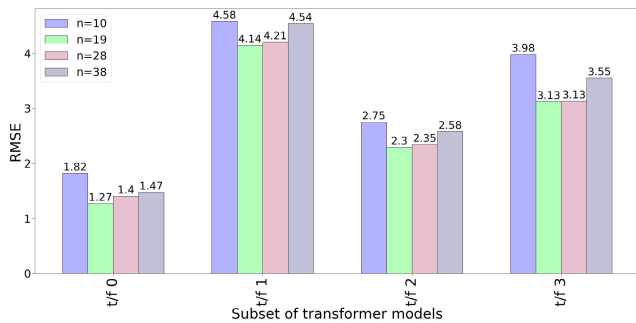


FIGURE 12. RMSE for increasing number (n) of features.

performance is obtained when the number of features is optimal ($n = 19$). If the number of features whilst training is increased or decreased, then the accuracy decreases.

D. FORECASTING RESULTS

The data obtained consisted of energy consumption values for 33 months for 1000 transformers. The K-Medoid clustering is utilized to cluster similar transformers together. The similarity indicates that transformers have similar patterns of aggregated daily consumption and hourly consumption. The aim of the work is to evaluate the performance of individual models for 1000 transformers against the clustered models. Individual models mean that 1000 transformers have separately trained models using their individual data i.e. these have 1000 different models. The clustered models indicate that the 1000 transformers are clustered into ‘ k ’ different groups and these ‘ k ’ clusters have one trained model each

trained on the transformers within the clusters. The employment of a clustering technique reduces the number of models required from 1000 to k . As described in the previous subsection, the value of ‘ k ’ (number of clusters) is optimized to minimize the within-cluster sum of square errors.

The performance of clustered and individual forecasting models for distribution transformers is evaluated in terms of RMSE, MAPE, training time, and testing time.

The RMSE and MAPE values for individual models and clustered models using DNNs are determined and these are depicted in Fig. 13 and Fig. 14. Fig. 13 indicates the results of the DNN models for load forecasting. Each of the subfigures indicates a representative subset of 1000 transformers. As observed from the RMSE lines, mostly the individual models represent the lower boundary of the two lines. The RMSE values range between 0 and 30 kWh. These values are very low considering the range (0 to 2,147,484 kWh) of energy consumption in the dataset. At a few points, the clustered models over perform the individual models for the respective transformers. The MAPE values for the individual models range between 4 to 16 percent and the MAPE for clustered models range between 5 to 19 percent. These MAPE values indicate that the clustered models are very comparable to the individual models. A few transformers exhibit high statistical variance in the energy consumption, i.e. they have either zero consumption values, or very high energy consumption values, or actual energy values range between 0 and 1. The MAPE values for such transformers are around 20-32%. These transformers have been found to be alternate backup transformers that are used only during the periods of faults, preventive, or predictive maintenance of main transformers.

Table 3 presents the results of clustering and individual models on dataset 1 when the machine learning models used are LR, LSTM, and DNNs. When accuracy is considered, the best performing model is the DNN model. In the clustering-based algorithm, the models are trained on a cluster whilst the testing is performed on each transformer within the cluster. If the clustering and individual models are compared, the individual models have slightly better accuracy when compared to clustered models. However, the accuracy of clustered models is highly competitive. If the gain on training time is considered, then the clustered models are highly preferable to individual models. When the training times for different machine learning models are considered, LR is the best owing to its simplicity. The DNN models have 10 folds of lesser training times compared to LSTM models. As a trade-off between accuracy and training time, it can be concluded that the clustering-based DNNs perform better. A similar pattern is also recognized in Table 4. It depicts the results of clustering and individual models on dataset 2 when LR, LSTM, and DNNs are used for training and testing.

The comparison of a trained clustered STLF model using different machine learning algorithms is illustrated in Fig. 15. The independent axis represents the time points and the dependent axis represents the energy consumption in kWh.

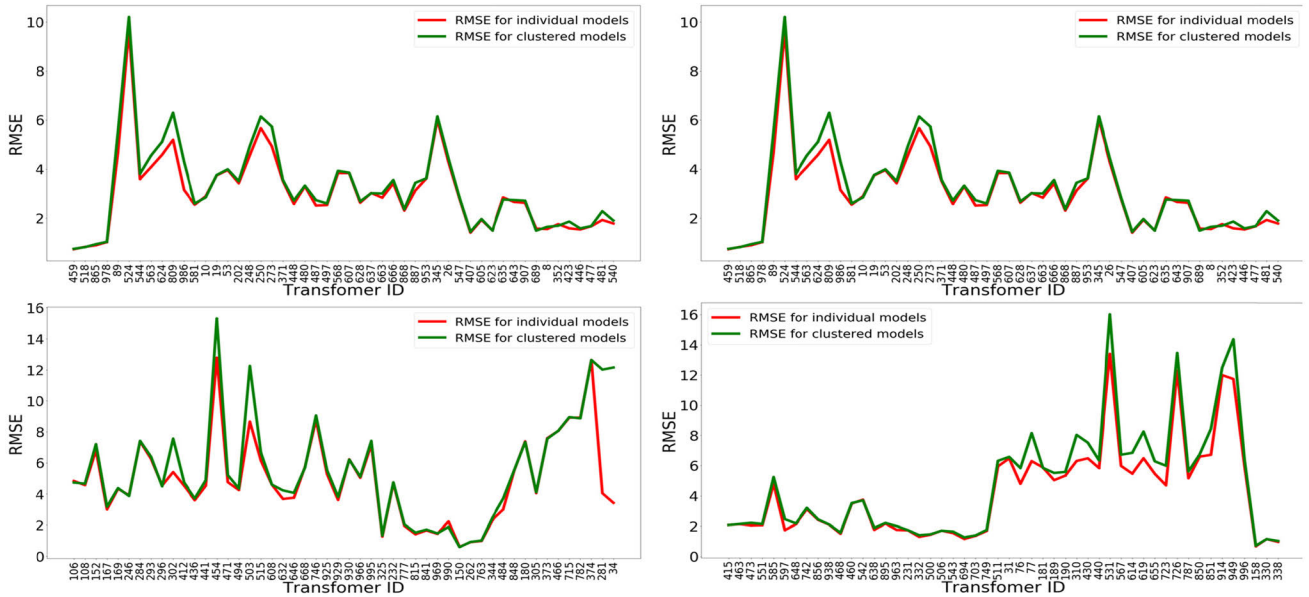


FIGURE 13. Performance evaluation (in RMSE) of clustered models v/s individual models for different transformers.

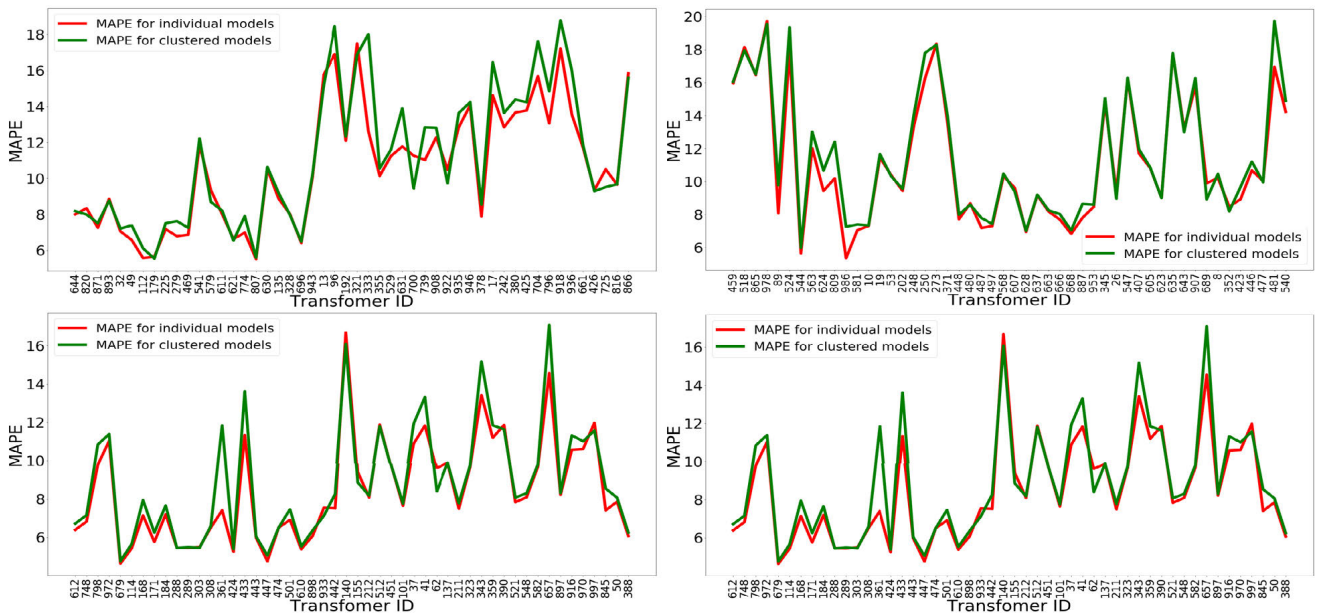


FIGURE 14. Performance evaluation (in MAPE) of clustered models v/s individual models for different transformers.

The results in the figure denote that the proposed k-Medoids methodology has generated accurate clusters, and the clustered model predicts energy consumption values close to the actual values of consumption for all machine learning algorithms in general. Fig. 15 also indicates that the DNN forecasts follow the consumption peaks better than LSTM and LR models. LSTM and LR at many time points forecast peaks after the peaks have occurred. Fig. 16 illustrates the error bars that depict the standard deviation of predictions using DNN and LSTM-based clustering models for STLF. The shaded

region around the blue line depicting predicted energy values using clustered DNN model represents the error region or the deviation of model predictions. The experiments were repeated a reasonable number of times i.e. 20 times to obtain the mean prediction and standard deviation of the predictions. LR-based clustering models had zero variance for predictions and hence, are not plotted. LSTM-based clustering models have variance tending to zero and additionally, DNN-based models have very low variance as shown in Fig. 16. The sources of randomness are kept at the minimum whilst

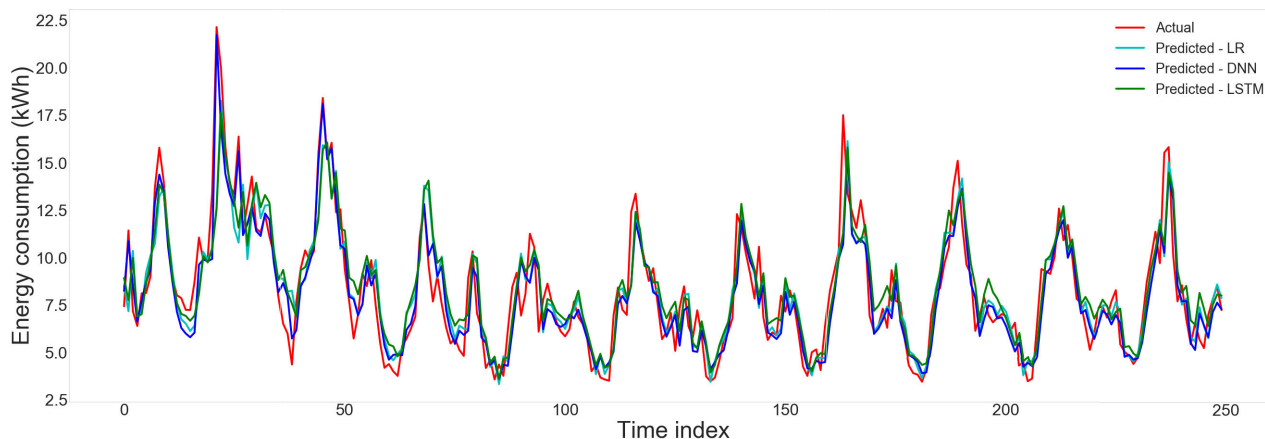


FIGURE 15. Forecast comparison of a trained clustered STLF model using different machine learning algorithms.

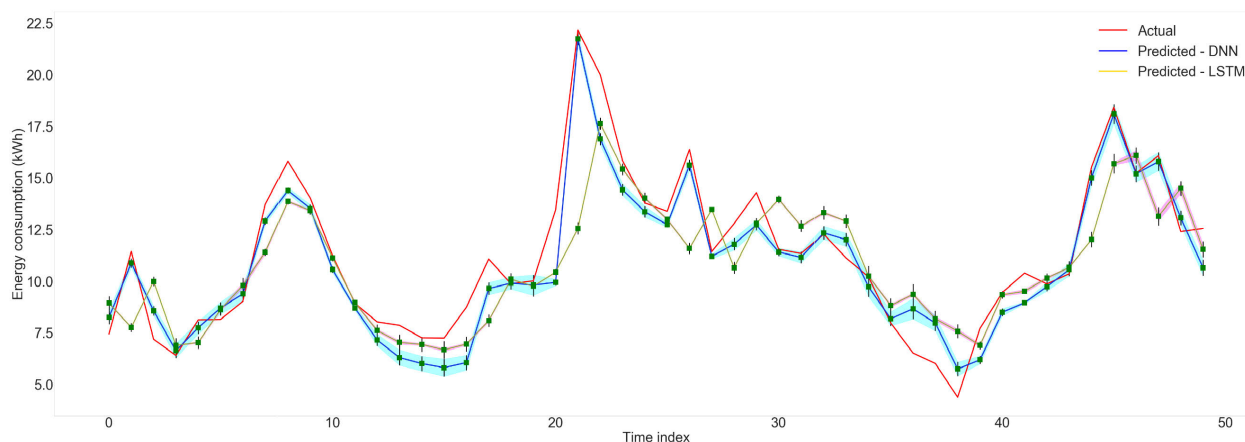


FIGURE 16. Error bars for forecasts using DNN and LSTM-based clustered STLF models.

TABLE 3. Results on dataset 1.

Model	Mean RMSE (kWh)	Mean MAPE (%)	Training Time (s)	Testing Time (s)
LR non-clustered	12.27	28.82	0.0656	0.0076
LR + clustering	13.25	32.23	0.0525	0.0076
ARIMA non-clustered	12.6305	30.2368	1.7182	1.8768
ARIMA + Clustering	14.2755	31.6923	1.0611	1.0047
LSTM non-clustered	2.2087	19.0902	421.00	3.7731
LSTM + clustering	3.1301	21.6020	118.83	0.2520
DNN non-clustered	2.3769	14.6451	14.63	0.0710
DNN + clustering	2.6874	15.9380	10.76	0.1070

training the proposed models and the trained models can be saved using deep learning serialization for future testing in industrial applications. The standard

deviation of the error metrics for retraining of forecasting models under similar initialization conditions will be negligible.

TABLE 4. Results on dataset 2.

Model	Mean RMSE (kWh)	Mean MAPE (%)	Training Time (s)	Testing Time (s)
LR non-clustered	54.0449	20.3235	14.06	0.89
LR + clustering	62.3503	20.8479	17.35	1.07
ARIMA non-clustered	59.4144	37.7886	513.5092	14.7560
ARIMA + Clustering	67.8725	39.4502	317.1253	8.6468
LSTM non-clustered	22.5221	7.2773	113432	378.86
LSTM + clustering	37.0626	11.1064	29106.35	110.41
DNN non-clustered	19.8216	7.1840	8409.14	6.08
DNN + clustering	21.2596	7.2271	4644.82	4.57

V. CONCLUSION

The electric utilities rely on load forecasting for capacity planning, power management, and operations in this era of uncertainty due to renewables integration. In this paper, a hybrid model of k-Medoids clustering and deep learning models for the day-ahead hourly load forecasting at the level of distribution transformers was proposed. The performance and applicability of this solution were demonstrated on two real datasets, which proves the generalization ability of the work. In the larger dataset 2, there were 1000 distribution transformers analyzed. These transformers were clustered based on the pairwise Minkowski similarity of aggregated daily consumption of energy using the k-Medoids clustering method with WCSSE measure. The elbow method determined that the optimum number of clusters for these 1000 transformers was 93. The reduction in the number of required models from 1000 to 93 reduced the constraint on the computational resources utilized for load forecasting and was a step towards real-time application. Consequently, the deep learning models were used to train and test the clustered models. These models were also compared with the 1000 individual deep learning models in the metrics of RMSE, MAPE, training time, and testing time values. The forecasting results indicated that the proposed methodology has generated accurate clusters and saved 44% of training time. In essence, the clustered models were highly competent with the individual models in terms of accuracy. Furthermore, the training time for clustered models was significantly lower than the individual models by a huge margin. The proposed methodology can be used with huge electrical networks and big data in smart grids at any level. Future work will investigate the scalability of the proposed methodology to a large electrical distribution network of 100,000 transformers for which the data are being collected currently.

ACKNOWLEDGMENT

The Open Access Funding was provided by Qatar National Library.

REFERENCES

- [1] S. M. Kaplan, *Smart Grid: Modernizing Electric Power Transmission and Distribution; Energy Independence, Storage and Security; Energy Independence and Security Act of 2007 (EISA); Improving Electrical Grid Efficiency, Communication, Reliability, and Resiliency; Integra*, 1st ed. The Capitol Net Inc, Alexandria, VA, USA, 2009.
- [2] Y. Hong, Y. Zhou, Q. Li, W. Xu, and X. Zheng, "A deep learning method for short-term residential load forecasting in smart grid," *IEEE Access*, vol. 8, pp. 55785–55797, 2020, doi: [10.1109/access.2020.2981817](https://doi.org/10.1109/access.2020.2981817).
- [3] B. Stephen, A. J. Mutanen, S. Galloway, G. Burt, and P. Järventausta, "Enhanced load profiling for residential network customers," *IEEE Trans. Power Deliv.*, vol. 29, no. 1, pp. 88–96, Feb. 2014, doi: [10.1109/tpwr.2013.2287032](https://doi.org/10.1109/tpwr.2013.2287032).
- [4] Y. Wang, Q. Chen, T. Hong, and C. Kang, "Review of smart meter data analytics: Applications, methodologies, and challenges," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 3125–3148, May 2019, doi: [10.1109/TSG.2018.2818167](https://doi.org/10.1109/TSG.2018.2818167).
- [5] J. Kwac, C.-W. Tan, N. Sintov, J. Flora, and R. Rajagopal, "Utility customer segmentation based on smart meter data: Empirical study," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, Vancouver, BC, Canada, Oct. 2013, pp. 720–725, doi: [10.1109/smartgridcomm.2013.6688044](https://doi.org/10.1109/smartgridcomm.2013.6688044).
- [6] M. E. H. Dyson, S. D. Borgeson, M. D. Tabone, and D. S. Callaway, "Using smart meter data to estimate demand response potential, with application to solar energy integration," *Energy Policy*, vol. 73, pp. 607–619, Oct. 2014, doi: [10.1016/j.enpol.2014.05.053](https://doi.org/10.1016/j.enpol.2014.05.053).
- [7] K. Hopf, M. Sodenkamp, and T. Staake, "Enhancing energy efficiency in the residential sector with smart meter data analytics," *Electron. Markets*, vol. 28, no. 4, pp. 453–473, Nov. 2018, doi: [10.1007/s12525-018-0290-9](https://doi.org/10.1007/s12525-018-0290-9).
- [8] S. J. Matthews and A. St. Leger, "Leveraging mapreduce and synchrophasors for real-time anomaly detection in the smart grid," *IEEE Trans. Emerg. Topics Comput.*, vol. 7, no. 3, pp. 392–403, Jul. 2019, doi: [10.1109/TETC.2017.2694804](https://doi.org/10.1109/TETC.2017.2694804).
- [9] D. Ilic, P. G. Da Silva, S. Karnouskos, and M. Griesemer, "An energy market for trading electricity in smart grid neighbourhoods," in *Proc. 6th IEEE Int. Conf. Digit. Ecosystems Technol. (DEST)*, Campione d'Italia, Italy, Jun. 2012, pp. 1–6, doi: [10.1109/dest.2012.6227918](https://doi.org/10.1109/dest.2012.6227918).
- [10] H. Guo, J. Liu, and L. Zhao, "Big data acquisition under failures in FiWi enhanced smart grid," *IEEE Trans. Emerg. Topics Comput.*, vol. 7, no. 3, pp. 420–432, Jul. 2019, doi: [10.1109/tetc.2017.2675911](https://doi.org/10.1109/tetc.2017.2675911).
- [11] D. Syed, A. Zainab, S. S. Refaat, H. Abu-Rub, and O. Bouhali, "Smart grid big data analytics: Survey of technologies, techniques, and applications," *IEEE Access*, early access, Nov. 27, 2020, doi: [10.1109/access.2020.3041178](https://doi.org/10.1109/access.2020.3041178).
- [12] L. Kuang, F. Hao, L. T. Yang, M. Lin, C. Luo, and G. Min, "A tensor-based approach for big data representation and dimensionality reduction," *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 3, pp. 280–291, Sep. 2014, doi: [10.1109/TETC.2014.2330516](https://doi.org/10.1109/TETC.2014.2330516).

- [13] C. L. Stergiou, K. E. Psannis, and B. B. Gupta, "IoT-based big data secure management in the fog over a 6G wireless network," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5164–5171, Apr. 2021, doi: 10.1109/jiot.2020.3033131.
- [14] M. Uddin, M. F. Romlie, M. F. Abdullah, S. Abd Halim, A. H. Abu Bakar, and T. Chia Kwang, "A review on peak load shaving strategies," *Renew. Sustain. Energy Rev.*, vol. 82, pp. 3323–3332, Feb. 2018, doi: 10.1016/j.rser.2017.10.056.
- [15] S. Refaat and H. Abu-Rub. (2015). *Implementation of Smart Residential Energy Management System for Smart Grid*. Accessed: Mar. 20, 2021. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/7310145/>
- [16] X. Wang, W.-J. Lee, H. Huang, R. L. Szabados, D. Y. Wang, and P. Van Olinda, "Factors that impact the accuracy of clustering-based load forecasting," *IEEE Trans. Ind. Appl.*, vol. 52, no. 5, pp. 3625–3630, Sep. 2016, doi: 10.1109/TIA.2016.2558563.
- [17] N. Huang, W. Wang, S. Wang, J. Wang, G. Cai, and L. Zhang, "Incorporating load fluctuation in feature importance profile clustering for day-ahead aggregated residential load forecasting," *IEEE Access*, vol. 8, pp. 25198–25209, 2020, doi: 10.1109/ACCESS.2020.2971033.
- [18] E. Paparoditis and T. Sapatinas, "Short-term load forecasting: The similar shape functional time-series predictor," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 3818–3825, Nov. 2013, doi: 10.1109/tpwrs.2013.2272326.
- [19] W. Kong, Z. Y. Dong, D. J. Hill, F. Luo, and Y. Xu, "Short-term residential load forecasting based on resident behaviour learning," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 1087–1088, Jan. 2018, doi: 10.1109/TPWRS.2017.2688178.
- [20] T. Vantuch, A. G. Vidal, A. P. Ramallo-Gonzalez, A. F. Skarmeta, and S. Misak, "Machine learning based electric load forecasting for short and long-term period," in *Proc. IEEE 4th World Forum Internet Things (WF-IoT)*, Singapore, Feb. 2018, pp. 511–516, doi: 10.1109/WF-IoT.2018.8355123.
- [21] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 841–851, Jan. 2019, doi: 10.1109/TSG.2017.2753802.
- [22] K. Chen, K. Chen, Q. Wang, Z. He, J. Hu, and J. He, "Short-term load forecasting with deep residual networks," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3943–3952, Jul. 2019, doi: 10.1109/TSG.2018.2844307.
- [23] S. Singh and A. Yassine, "Mining energy consumption behavior patterns for households in smart grid," *IEEE Trans. Emerg. Topics Comput.*, vol. 7, no. 3, pp. 404–419, Jul. 2019, doi: 10.1109/TETC.2017.2692098.
- [24] Z. Han, M. Cheng, F. Chen, Y. Wang, and Z. Deng, "A spatial load forecasting method based on DBSCAN clustering and NAR neural network," in *Proc. J. Phys., Conf.*, 2020, vol. 1449, no. 1, Art. no. 012032, doi: 10.1088/1742-6596/1449/1/012032.
- [25] H. H. H. Aly, "A proposed intelligent short-term load forecasting hybrid models of ANN, WNN and KF based on clustering techniques for smart grid," *Electr. Power Syst. Res.*, vol. 182, May 2020, Art. no. 106191, doi: 10.1016/j.epr.2019.106191.
- [26] Q. Zhang and J. Zhang, "Short-term load forecasting method based on EWT and IDBSCAN," *J. Electr. Eng. Technol.*, vol. 15, no. 2, pp. 635–644, Mar. 2020, doi: 10.1007/s42835-020-00358-0.
- [27] B. Nepal, M. Yamaha, A. Yokoe, and T. Yamaji, "Electricity load forecasting using clustering and ARIMA model for energy management in buildings," *Jpn. Architectural Rev.*, vol. 3, no. 1, pp. 62–76, Jan. 2020, doi: 10.1002/2475-8876.12135.
- [28] F. Fahiman, S. M. Erfani, and C. Leckie, "Robust and accurate short-term load forecasting: A cluster oriented ensemble learning approach," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–8.
- [29] P. Bento, J. Pombo, S. Mariano, and M. D. R. Calado, "Short-term load forecasting using optimized LSTM networks via improved bat algorithm," in *Proc. Int. Conf. Intell. Syst. (IS)*, Frankfurt, Germany, Sep. 2018, pp. 351–357, doi: 10.1109/IS.2018.8710498.
- [30] D. Syed, H. Abu-Rub, A. Ghayeb, and S. S. Refaat, "Household-level energy forecasting in smart buildings using a novel hybrid deep learning model," *IEEE Access*, vol. 9, pp. 33498–33511, 2021, doi: 10.1109/ACCESS.2021.3061370.
- [31] L. Wang, S. Mao, and B. Wilamowski, "Short-term load forecasting with LSTM based ensemble learning," in *Proc. Int. Conf. Internet Things (iThings) IEEE Green Comput. Commun. (GreenCom) IEEE Cyber. Phys. Social Comput. (CPSCoM) IEEE Smart Data (SmartData)*, Atlanta, GA, USA, Jul. 2019, doi: 10.1109/iThings/GreenCom/CPSCoM/SmartData.2019.00145.
- [32] D. Syed, S. S. Refaat, H. Abu-Rub, O. Bouhali, A. Zainab, and L. Xie, "Averaging ensembles model for forecasting of short-term load in smart grids," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Los Angeles, CA, USA, Dec. 2019, pp. 2931–2938, doi: 10.1109/Big-Data4790.2019.9006183.
- [33] G. Sideratos, A. Ikonopoulos, and N. D. Hatzigiorgiou, "A novel fuzzy-based ensemble model for load forecasting using hybrid deep neural networks," *Electr. Power Syst. Res.*, vol. 178, Jan. 2020, Art. no. 106025, doi: 10.1016/j.epr.2019.106025.
- [34] S. Dasgupta, A. Srivastava, J. Cordova, and R. Arghandeh, "Clustering household electrical load profiles using elastic shape analysis," in *Proc. IEEE Milan PowerTech*, Milano, Italy, Jun. 2019, pp. 1–6, doi: 10.1109/PTC.2019.8810883.
- [35] L. C. P. Velasco, N. R. Estoperez, Renbert J. R. Jayson, C. J. T. Sabijon, and V. C. Sayles, "Day-ahead base, intermediate, and peak load forecasting using K-Means and artificial neural networks," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 2, pp. 62–67, 2018, doi: 10.14569/ijacsa.2018.090210.
- [36] F. L. Quilumba, W.-J. Lee, H. Huang, D. Y. Wang, and R. L. Szabados, "Using smart meter data to improve the accuracy of intraday load forecasting considering customer behavior similarities," *IEEE Trans. Smart Grid*, vol. 6, no. 2, pp. 911–918, Mar. 2015, doi: 10.1109/tsg.2014.2364233.
- [37] S. A. Iberdrola. *STAR Project: A Global Benchmark in Smart Grids Investments*. Accessed: Dec. 31, 2020. [Online]. Available: <https://www.iberdrola.com/about-us/lines-business/flagship-projects/star-project>
- [38] *Dark Sky Api-Weather Conditions*. Accessed: Dec. 31, 2020. [Online]. Available: <https://darksky.net/dev>
- [39] J. Sessa and D. Syed, "Techniques to deal with missing data," in *Proc. 5th Int. Conf. Electron. Devices, Syst. Appl. (ICEDSA)*, Sarawak, Malaysia, Dec. 2016, pp. 1–4, doi: 10.1109/ICEDSA.2016.7818486.
- [40] K. P. Sinaga and M.-S. Yang, "Unsupervised K-means clustering algorithm," *IEEE Access*, vol. 8, pp. 80716–80727, 2020, doi: 10.1109/ACCESS.2020.2988796.
- [41] R. T. Ng and J. Han, "CLARANS: A method for clustering objects for spatial data mining," *IEEE Trans. Knowl. Data Eng.*, vol. 14, no. 5, pp. 1003–1016, Sep. 2002, doi: 10.1109/tkde.2002.1033770.
- [42] C. M. Bishop, *Pattern Recognition and Machine Learning*, 1st ed. New York, NY, USA: Springer, 2006.
- [43] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 2016.
- [44] D. Li, L. Deng, B. Bhooshan Gupta, H. Wang, and C. Choi, "A novel CNN based security guaranteed image watermarking generation scenario for smart city applications," *Inf. Sci.*, vol. 479, pp. 432–447, Apr. 2019, doi: 10.1016/j.ins.2018.02.060.
- [45] Y. Li and Y. Liang, "Learning overparameterized neural networks via stochastic gradient descent on structured data," in *Proc. Adv. Neural Inf. Process. Syst.*, Montreal, QC, Canada, 2018, pp. 8157–8166. [Online]. Available: <https://proceedings.neurips.cc/paper/2018/file/54fe976ba170c19ebae453679b362263-Paper.pdf>
- [46] S. R. Sahoo and B. B. Gupta, "Multiple features based approach for automatic fake news detection on social networks using deep learning," *Appl. Soft Comput.*, vol. 100, Mar. 2021, Art. no. 106983, doi: 10.1016/j.asoc.2020.106983.
- [47] A. Zainab, D. Syed, A. Ghayeb, H. Abu-Rub, S. S. Refaat, M. Houchati, O. Bouhali, and S. Banales Lopez, "A multiprocessing-based sensitivity analysis of machine learning algorithms for load forecasting of electric power distribution system," *IEEE Access*, vol. 9, pp. 31684–31694, 2021, doi: 10.1109/ACCESS.2021.3059730.
- [48] M. A. Syakur, B. K. Khotimah, E. M. S. Rochman, and B. D. Satoto, "Integration K-means clustering method and elbow method for identification of the best customer profile cluster," in *Proc. IOP Conf., Mater. Sci. Eng.*, Phuket, Thailand, 2018, vol. 336, no. 1, p. 12017, doi: 10.1088/1757-899x/336/1/012017.
- [49] F. Batool and C. Hennig, "Clustering with the average silhouette width," *Comput. Statist. Data Anal.*, vol. 158, Jun. 2021, Art. no. 107190, doi: 10.1016/j.csda.2021.107190.
- [50] S. K. Kingrani, M. Levene, and D. Zhang, "Estimating the number of clusters using diversity," *Artif. Intell. Res.*, vol. 7, no. 1, pp. 15–22, 2018, doi: 10.5430/air.v7n1p15.
- [51] J. Robertson and M. Kaptein, "An introduction to modern statistical methods in HCL," in *Modern Statistical Methods for HCI*, 1st ed. New York, NY, USA: Springer, 2016, pp. 1–14.
- [52] C. Yuan and H. Yang, "Research on K-value selection method of K-means clustering algorithm," *J.*, vol. 2, no. 2, pp. 226–235, Jun. 2019, doi: 10.3390/j2020016.



DABEERUDDIN SYED (Member, IEEE) received the Bachelor of Engineering degree in electrical and electronics engineering (EEE) from the University College of Engineering, Osmania University, India, in 2013, and the M.Sc. degree in data science and engineering from Hamad Bin Khalifa University, Qatar, in 2018. He is currently pursuing the Ph.D. degree in electrical and computer engineering (ECEN) with Texas A&M University (TAMU), College Station, TX, USA. He has industrial work experience of three years. He started his career as a Database Administrator with Oracle Pvt. Ltd. He then attained experience and skills as a Research Assistant and as a Teaching Assistant on multiple occasions. He has previously assisted in teaching courses, such as electrical circuits, learning from data, and introduction to data science with python programming. He has published over 18 research articles and has one book chapter to his credit. His research interests include smart grids, big data analytics, load forecasting, and distributed computing.



HAITHAM ABU-RUB (Fellow, IEEE) is currently a Full Professor with two Ph.D. degrees. He attained research and teaching experience in diverse countries of Germany, Palestine, Poland, Qatar, and USA. He has been working with TAMU at Qatar (TAMUQ), for the past 17 years. He has worked as a Chair of the ECEN Program, TAMUQ, for a period of five years. He is the Managing Director (MD) of the Smart Grid Center, Qatar (SGCQ). He has published over 400 research articles and has five books, and six book chapters to his credit. His main research interests include renewable energy, power electronic converters, smart grid, and electric drives. He was a recipient of the German Alexander von Humboldt Fellowship, and the American Fulbright Scholarship. He received numerous recognitions, international and national awards.



ALI GHRAYEB (Fellow, IEEE) received the Ph.D. degree in electrical engineering (EE) from the University of Arizona, USA. He has worked as a Professor with the Department of ECEN, Concordia University, Canada. He is currently a Professor with the Department of ECEN, TAMUQ. His research interests include massive MIMO, wireless communications, physical layer security, and visible light communications. He has served on the editorial board of diverse IEEE and non-IEEE journals.



SHADY S. REFAAT (Senior Member, IEEE) received the B.A.Sc., M.A.Sc., and Ph.D. degrees in electrical engineering from Cairo University, Egypt, in 2002, 2007, and 2013, respectively. He attained industrial work experience of over 13 years. He worked as a Senior Electrical Engineer, a Team Leader, and a Design Engineer on multiple engineering projects. He currently works as an Associate Research Scientist with TAMUQ. He has published over 105 research articles. His research interests include big data, condition monitoring, electrical machines, energy management systems, fault detection, power systems and reliability, fault-tolerant systems, and smart grid. He is serving as a member of the SGCQ.



MAHDI HOUCHATI (Member, IEEE) received the B.Sc. degree in electrical engineering from the National School of Engineering of Sfax, Tunisia, and the M.Sc. degree from Qatar University (QU). He is currently a Senior Engineer with Iberdrola QSTP with more than 11 years of experience in industrial companies and research institutes. Through the variability of the roles he was carrying, he attained knowledge in modeling, optimization, energy management, machine learning, and systems design. Along with his experience in the industrial sector where he was involved in the development and prototyping of a new industrial product, he was involved in several research topics as a Research Assistant with QU and Qatar Environment and Energy Research Institute. During the last eight years, he worked on multiple electrical and multidisciplinary projects involving machine learning, building audits, energy efficiency, energy management, and big data.



OTHMANE BOUHALI (Member, IEEE) is currently a Physics Research Professor with TAMUQ. He is also the Founder and the Director of the TAMU Advanced Scientific Computing Center. He has been involved in the Large Hadron Collider research program for over 26 years and has supervised numerous research projects. His research interests include large-scale modeling, medical physics, high-performance computing, and detector technologies for radiation.



SANTIAGO BAÑALES (Member, IEEE) received the M.Sc. degree from MIT (Fulbright Fellow) and a Dual Engineering Degree from the Ecole Centrale Paris, Universidad Politecnica de Madrid. He has experience of over 21 years in the energy industry. Over the years, he has fulfilled executive roles in Research and Development, Engineering, Management Consulting, Product Development, M&A, and General Management (CTO and CEO). He has experienced a wide breadth of business environments in an international setting, doing business with state-owned companies, partnership firms, private equities, and family-owned manufacturing businesses. He has realized executive and non-executive Board Directorships assignments in several energy industry associations and institutions and has been involved in graduate and executive education for over 11 years. He is currently the Managing Director of Iberdrola Innovation Middle East, Iberdrola's Digital Utility Innovation Centre based in Qatar.

...