Research Article

Academic Platform Journal of Engineering and Smart Systems (APJESS) 11(3), 151-162, 2023

Received: 16-Mar-2023 Accepted: 28-Aug-2023

homepage: https://dergipark.org.tr/tr/pub/apjess

https://doi.org/10.21541/apjess.1266610

A Novel Clustering-based Forecast Framework: The Clusters with Competing Configurations Approach

[1]Miray ALP, [*2]Gökhan DEMİRKIRAN

[1] Department of Electrical and Electronics Engineering, Yaşar University, İzmir, Türkiye, mirayalp94@gmail.com

[2] Department of Electrical and Electronics Engineering, Yaşar University, İzmir, Türkiye, gokhan.demirkiran@yasar.edu.tr

Abstract

Accurate aggregate (total) short-term load forecasting of Smart Homes (SHs) is essential in planning and management of power utilities. The baseline approach consists of simply designing and training predictors for the aggregated consumption data. Nevertheless, better performance can be achieved by using a clustering-based forecasting strategy. In such strategy, the SHs are grouped according to some metric and the forecast of each group's total consumption are summed to reach the forecast of aggregate consumption of all SHs. Although the idea is simple, its implementation requires fine-detailed steps. This paper proposes a novel clustering-based aggregate-level forecast framework, so called Clusters with Competing Configurations (CwCC) approach and then compares its performance to the baseline strategy, namely Clusters with the Same Configurations (CwSC) approach. The configurations in the name refer to the configurations of ARIMA, Multi-Layer Perceptron (MLP), and Long Short-Term Memory (LSTM) forecasting methods, which the CwCC approach uses. We test the CwCC approach on Smart Grid Smart City Dataset. The results show that better performance can be achieved using the CwCC approach for each of the three forecast methods, and LSTM outperforms other methods in each scenario.

Keywords**:** clustering; deep neural networks; short term load forecasting; smart grid.

1. INTRODUCTION

Energy is regarded as the most crucial aspect of a region's social and economic growth since it significantly contributes to its progress and improved economy. Concerns about its uninterrupted supply especially to the vital operations arose as the electricity consumption has risen dramatically in recent years, primarily due to population, home, and commercial floor space increases [1]. While installing back-up power grid infrastructures is one (costly) option, the regulation of consumption through demand response programmes seems one practical and cheap way to maintain the supply-demand balance by restricting the demand (The term demand is interchangeably used for the consumption or the load) to the supply. One strategy to incentivize customers to reduce consumption to meet supply is to use reward-based response programs [2]. Reward-based programs can also reduce operating costs of grids, improving the economic conditions of both providers and consumers [3].

Operations of demand response programmes in modern power systems generally require the information of future load consumption over an interval of 1-hour to 1-week (short-term load forecasting) which may be at the aggregate level or the residential level. For example, in one implementation scenario of the demand response programme, the utility operator forecasts the aggregated residential load (i.e., total load) in an area and decides if an action is needed to be taken based on that aggregated load forecast [4]. Simultaneously, it can also use algorithms based on the individual customer data (such as load profiles and forecasts of individual customer consumptions) to evaluate the potential of the customers to participate the program, and design accordingly the specification of the demand response to be initiated. Apparently, in such implementations the success of the demand response programmes relies on the accuracy of the forecasts. Having said that, in this study, our aim is to increase the performance of aggregate-level forecasting. Our strategy is to cluster the aggregate consumption into clusters of similar and better-forecastable consumption profiles and then sum the forecast of each group to reach the forecast of aggregate consumption.

Mostly clustering is used as a tool to comprehend user profiles to be used in demand response scenarios. For example, time series clustering can model consumer behavior for decision-making in Swedish electricity market that is subject to ongoing developments [5]. The diversity of user profiles in Swish electricity market is inevitable as shown by [6] via comparing different clustering approach.

Moreover, Reference [7] emphasizes the need for using diverse profiles instead of relying on one standard pattern for policy makers. One challenge with consumer behaviour is that the user load profiles are changeable, thereby diminishing the efficiency of static load profiling approaches. In this direction, a recursive clustering algorithm that continuously update the load profiles using newly updated data has been proposed by [8]. Load profile classification can also be improved by a spectral clustering algorithm [9]. Despite the efforts for more accurate load profiling, the references [5]–[9] do not address how to utilize such load profiling in forecasting. A customizable toolbox Divinus which uses clustering methods both for user profiling and for forecasting total consumption is proposed by [10]. However, they use clustering to forecast by remembering the previous data in a non-generalizable way. Clustering is also used to increase the performance of individual residential load forecasting, since clustered load profiles in the cluster which the residential belongs to are similar, thus can be used increase the data diversity of individual load forecasting and decrease the overfitting [11], [12].

Table 1. The references using cluster-based aggregate-level forecasting.

Reference	Dataset	Total number of houses	Forecast Method	Inputs (Features)	Cluster Method	Clustering Objective
[13]	CER	6000	PARX	Load signal, Calendar, Temperature.	spectral clustering	similarity matrix
[14]	CER	3639	Linear Regression, MLP, SVR, ARIMA.	Load signal, calendar features.	Max-AC, Min-StDev, Max-Sim.	24-hour load profile
[15]	CER	3176	MLP	Load signal, Calendar features, Temperature.	k-means	Regression Coefficients, 24-hour load profile.
[16]	Enernoc, CER	782	SVM, MLP, Linear Regression, Ensemble.	Load signal	greedy algorithm	entire sequence
[17]	CER	3176	MLP, Deep Belief Network.	Load signal, Calendar features, Temperature.	k-shape	24-hour load profile
[18]	Slovak Electricity consumption	1152	LSTM	Load signal	k-shape	entire sequence
[19]	Arbon(Swiss), CER.	7500	LSTM variant	Load signal	spectral clustering	similarity matrix

Most load forecasting studies use point forecasts based on the aggregate system-level data classified as statistical approaches, machine learning techniques, or hybrid models [20]. However, the number of studies using clustering techniques to improve aggregate-level forecasting are relatively few. We summarize the most significant 7 papers, which share the same aim and strategy with this paper in Table-1. These papers experiment with clustering-based aggregate load forecasting, from different point of views. 6 of 7 papers use Commission for Energy Regulation (CER) dataset from Ireland whilst one of them uses Slovak Electricity Consumption. The total number of houses range from 782 to 7500. The forecasting algorithms used are MLP (Multi-Layer Perceptron), SVR (Support Vector Regression), ARIMA, Deep Belief Networks, LSTM (Long Short-Term Memory), Linear Regression Ensemble and PARX (Periodic Auto-Regressive model with Exogenous variables). As the input features, calendar and temperature information can be used together with the load signal itself, specifically in Artificial Intelligence (AI) techniques. The references in Table-1 use similarity matrix, 24-hour load profile, regression coefficients, or the entire sequence as the input to the clustering algorithms. Overall, in all the references in Table-1, clustering is shown to be an effective technique that increases the forecasting performance regardless of the clustering technique used, though some clustering techniques may give better forecasting performance [17].

However, the studies listed in Table-1 do not directly experiment with the strategy, rather use or modify it as a tool to tackle a particular issue. In this paper, we conceptualize this strategy as a standard framework, and further propose an immediate-update to it, so called Clusters with Competing Configurations, so that the clustered groups can have their own forecast structures that fits better to their load profile characteristics. These forecast structures can have different configurations of the same or a different methodology.

Regarding the methodologies, we conduct a comparison of ARIMA, MLP, and LSTM in clustering-based aggregate load forecasting. That is, we compare the most-adopted linear statistical method ARIMA with its widely used nonlinear AI counterparts, i.e. MLP and LSTM. MLP is a shallow AI technique, whilst LSTM is the state-of-the-art deep AI technique that has been very popular recently. A good configuration parameter of a naive method can have a better performance than a poorly configured state-of-the-art model. Thus, to make sure that the best possible performances of each forecasting algorithm is adopted in forecasting, we propose a clustering-based aggregate-level short-term load forecasting, so called Clusters with Competing Configurations (CwCC) approach. CwCC approach includes the configuration spaces of these algorithms, which the references in Table-1 lack, such as the different sets of lags in ARIMA, the number of neurons in MLP, and the different number of hidden units in LSTM etc. That is, in our strategy we have different configurations of the same forecasting method competing for each cluster's forecast. That is, there can be different configurations of the same forecasting method for different clusters. Additionally, to compare the effect of adding configuration space in CwCC approach, we test the case of using the same configuration (but different weights learned during training of each cluster) of the same forecasting method across all clusters, which we call as Clusters with the Same Configuration (CwSC) approach.

Furthermore, for a fair comparison, we use only the load signal as the input to the MLP and LSTM, as the ARIMA can only handle 1-dimensional data. As for the clustering input, we use 24-hour load profile, which is the most practical option compared to the similarity matrix that needs several user-tuned parameters, and to the entire sequence which is subject to the curse of dimensionality. Additionally, we provide clear steps and reproducible results for those who wish to implement the cluster-based strategy for real-life use.

Finally, we use Smart Grid Smart City dataset, which can provide 12641 residential load time series in the context of this study, that is more residentials than CER and Slovak Electricity Consumption datasets that references in Table-1 use, can provide. Our results indicate that CwCC approach is an effective technique to increase the aggregate-level forecast performance and LSTM models in general gives better performance in terms of MAPE.

The rest of this paper is organized as follows. Section 2 presents the elements of the proposed framework: k-means clustering, Long Short-Term Memory (LSTM), ARIMA, Multi-Layer Perceptron (MLP). Section 3 describes the SGSC dataset and analyzes different time scales' effect on energy consumption and patterns essential for modeling. Section 4 assesses the proposed method by evaluating the accuracy of forecasts. Finally, Chapter 5 presents conclusions and recommendations.

2. THE REVIEW OF THE ALGORITHMS

2.1. k-means Clustering

K-means clustering is a method of logically classifying raw data and searching for hidden patterns in datasets. It arranges data into fragmented (k-)clusters such that data in one cluster is identical to that in another, but data in other clusters differs. K-means is an iterative, numerical, unsupervised, and non-deterministic approach that assigns n observations to exactly one of the k clusters defined by centroids, where k is a predetermined number. It is straightforward and easy; hence it has shown to be a very useful strategy for producing good clustering results in many practical applications.

2.2. ARIMA

ARIMA is a computational iteration with polynomial structure in the form of:

$$x_t = c + x_{t-1} + \phi_1 x_{t-1} + \cdots + \phi_p x_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \quad (1)$$

Where ϕ_i refers to the autoregressive coefficients, θ_i to the moving average coefficients, c to the bias. $\varepsilon_t = x_t - \widehat{(x_t)}$ is the error which is assumed to be a Gaussian distribution with constant variance. The zero coefficients are unknown but can be determined using statistical analysis, such as autocorrelation (In other words, autoregressive and moving average lags are unknown, but can be inferred by autocorrelation). After that the exact values of non-zero coefficients and bias are learned during training.

In the following, we carry out an autocorrelation analysis of the data used in this paper (See Chapter 3.1 for data) to find the lags of non-zero ϕ_i and θ_i. Autocorrelation analysis in Fig.1 reveals that the consumption at a given time is more related to previous increments, and as well as the same time of the previous days. The correlation decreases as the number of previous days increases. In designing the configuration space in Table 2 which will be used in experimentation, we have used lag samples from the peaks of the sample autocorrelation function sketched in Fig. 1 and our pre-experiments. Specifically, we have designed an experiment from the lags around the peaks of 1st, 48th (24th hour or previous day) and 96th (48th hour or the previous two day) lag which represent the peak correlations to the previous samples of the data (Table 2).

Table-2. Configuration Space of ARIMA models.

Index	AR Lags	MA Lags
1	1 2	1 2
2	1 2 3	1 2 3
3	1 2 3 4	1 2 3 4
4	1 2 3 4	1 2 47 48 49
5	1 2 3 4	1 2 95 96 97
6	1 2 47 48 49	1 2 3 4
7	1 2 47 48 49	1 2 47 48 49
8	1 2 47 48 49	1 2 95 96 97
9	1 2 95 96 97	1 2 3 4
10	1 2 95 96 97	1 2 47 48 49
11	1 2 95 96 97	1 2 95 96 97
12	1 2 3 4 47 48 49	1 2 3 4
13	1 2 3 4 47 48 49	1 2 47 48 49
14	1 2 3 4 47 48 49	1 2 95 96 97
15	1 2 3 4 95 96 97	1 2 3 4
16	1 2 3 4 95 96 97	1 2 47 48 49
17	1 2 3 4 95 96 97	1 2 95 96 97

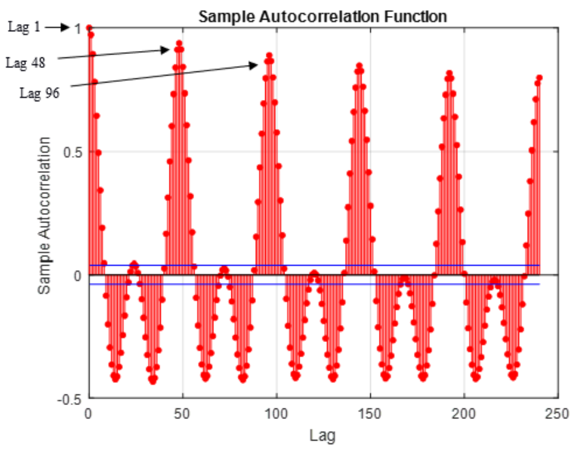

Figure 1. Autocorrelation analysis of data.

2.3. Multi-Layer Perceptron (MLP)

MLP is a type of feedforward neural network, which in fact uses a structured function with unknown parameters (weight parameters) to approximate a hypothetical function which perfectly maps (training) input to (training) outputs. MLPs are in the form of:

$$\sigma_n\left(W_n \dots \left(\sigma_1\left(W_1\left(\sigma_0(W_0 x + b_0)\right) + b_1\right)\right) + b_n\right) = y \qquad (2)$$

Where n is the number of hidden layers, $W_0, .. W_n$ and $b_0, \dots, b_n$ are the unknown weight matrices, where one of the dimensions of each weight matrix is determined empirically (the number of neurons), whereas the second dimension is determined by the rules of matrix multiplication. $\sigma_0, \dots, \sigma_n$ are the activation functions, which can be a tangent hyperbolic, a sigmoid, a linear function etc., again determined empirically. x is as the architecture space of MLP we have used one hidden layer whose number of neurons are taken from the set {5,10,15,20}. For the optimizer, we have used Levenberg-Marquardt as it outperformed other algorithms in general in our pre-experiments. Additionally, the two or more hidden layers greatly decreased the test performance, thus they are discarded from the configuration space.

2.4. Long Short-Term Memory (LSTM)

LSTM is a type of RNN, which can learn long-term dependencies. LSTM, initially proposed by [21] is a feed-forward neural network that unfolds in time (Fig. 2). The unknown matrices are learned during training process of LSTM by backpropagation through time algorithm, which prevents the vanishing gradient problem.

The configuration space of LSTM used in this paper are hidden unit layers (50,100,200,500), optimizers (adam, sgdm, rmsprop) and the number of epochs (500,1000,2000), which result in (4x3x3) 36 different configurations. Each of the configuration numbers from 1 to 36 consists of hyperparameters which must be empirically determined so that the best model can be found. In our pre-experiments, we have observed that the deviation of learning rate from 0.005 has negative effect on results in general. Thus, we fix it to 0.005. The design of the configuration space is needed several pre-experiments with data.

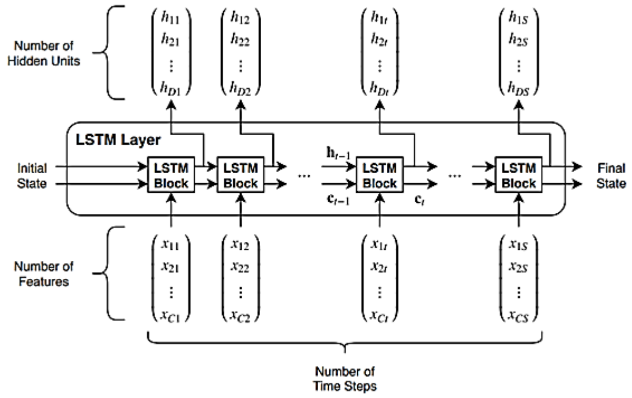

Figure 2. The Computational Architecture of LSTM in MATLAB®.

3. PREPROCESSING DATASET

3.1. Smart Grid Smart City Dataset

In this research, Smart Grid Smart City dataset from Australia [22] is used. The field where data is collected includes energy distribution, transmission businesses, technology firms, universities, and the CSIRO. Since 2009, SGSC project have been deployed in eight local government regions in New South Wales, covering 30,000 residences and serving as one of the world's largest commercial-scale smart grid technology assessment projects. Climate zone, household income, housing type, electricity consumption, and gas consumption level are used to segment the population, yielding 108 socio-demographic statistics cells in total [23].

SGSC data is obtained from 13,735 customers between the years 2010-2014. It includes comprehensive data on appliance use, climate, retail, and distributor product offers, and other associated elements, as well as one of the few linked sets of consumer time of use (with half-hour increments) and demographic data for Australia. Electricity consumption interval readings, home area network plug readings, peak events, peak events reaction, and offer and acceptance of the event signal are some of the data resources for this dataset.

3.2. Data Preprocessing

SGSC dataset is about 20 gigabyte volume of .csv file, which cannot be loaded into a conventional laptop due to ram restrictions. Thus, first we divide the data into .csv files of 1 gigabyte data so that it can be loaded to the ram and MATLAB can reach data. After that, we store 13,735 .mat files (MATLAB® storage file) containing the time series data of each house by carefully searching the houses in that 20 gigabyte data.

The electrical consumption data in SGSC dataset has half-hourly intervals. Additionally, the number of days of all recorded data were very different from household to other. For example, it has been seen that there are households with 2 days of data, as well as households with more than 800 days of data (Fig. 3).

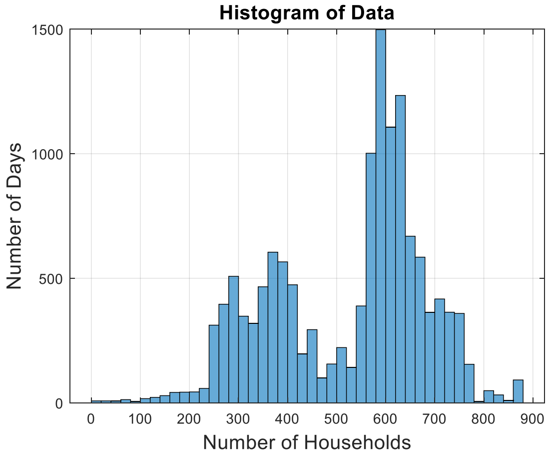

Figure 3. Histogram of number of days in the dataset.

Another important consideration when dealing with time series data is there can be missing as well as double-entered values, which is very problematic due to shifting in data. Thus, first we examine the dataset values according to the below Algorithm-1 and create a proper time series.

Algorithm 1. Fixing the missing and repeated measurements.

Data Preprocessing

```
Upload a data in the form of table
Find improper increments in the data
    if replicated is found
        Remove duplicate data from table
        Update the data as the mean of two-readings
        Start the algorithm from beginning
    end
    if missing is found
        Find the mean of the data before and after
        Replace the missing data
        Start the algorithm from beginning
    end
```

3.3. Finding a Proper Subset of Data

Figure 4 shows the datetime interval in which a customer participates to the SGSC program (blue dots) and leaves the program (red dots). We decide that having 8 weeks of data for training, 2 weeks for validation, and 2 weeks for testing is appropriate when compared to the studies in Table-1, thus 12-week duration of data is used. The interval date of 12-week duration is specifically chosen such that there will be the highest number of houses that can provide data of 12-week duration, by the help of Fig. 5. Fig.5 depicts the number of houses that can provide data on the interval of [min_starting date + Ithweek, min_starting date + Ithweek + Xweeks], where I(thweek) increases from 0 to last possible number searching for the date and Xweeks is the duration, which is 12weeks in our case. In Fig.5 also, other X-week durations are represented for comparison for the reader.

When searching for the interval date of 12-week duration, we also looked for electricity consumption in one season only to avoid seasonal factors. The highest number of houses, which is 12641, in one season using 12-week duration occurs in the interval date of [03-June-2013, 26-August-2013], i.e. [the first Monday of June, and the last Sunday of August] (summer in northern semi-sphere) (Fig. 4). The aggregation of 12641 household consumptions for train, validation and test data is depicted in Fig. 6.

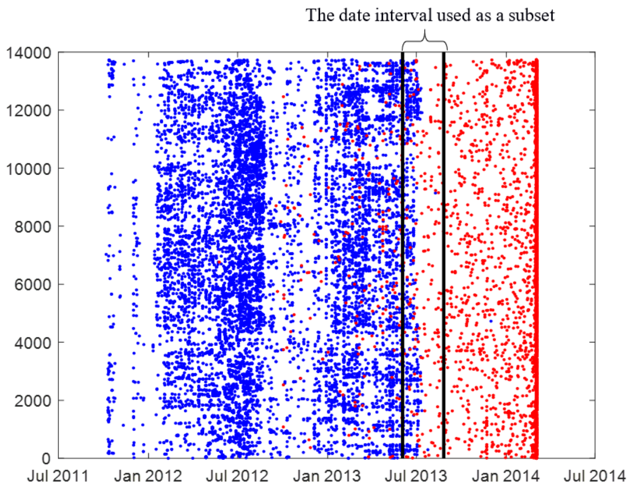

Figure 4. The starting and end date of households participating to SGSC program

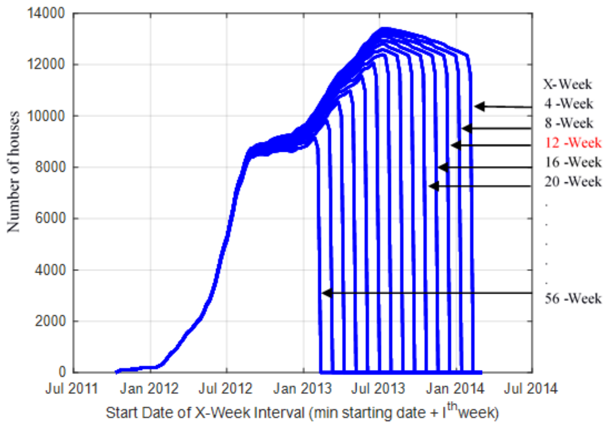

Figure 5. The moving Xweek interval vs. the number of houses that can provide data in that interval.

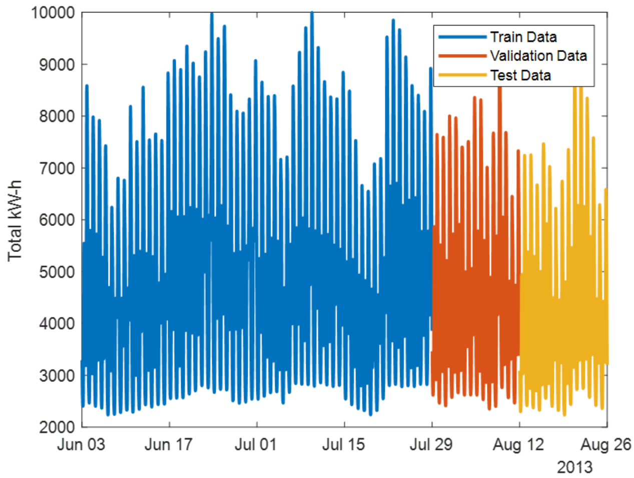

Figure 6. Total Consumption between [03-June-2013,26-August-2013] obtained by aggregating 12641 households.

3.4. k-means Clustering

In this subsection, we cluster the normalized consumption profiles of the households. The normalized consumption profiles are the mean of the daily consumptions for each measurement averaged over total 84 days of consumptions for each house (12 Weeks = 84 days). That is each house has a normalized load profile of a 1x48 (0.5 hour of incremental during day) vector, each element of vector corresponding to the average consumptions in that date interval. For clustering, we have experimented with different number of clusters, ranging from 2 to 10.

Fig. 7 shows an example of the total consumption of each cluster in case of 4 cluster used and the combined total consumption of all clusters. Fig. 8 depicts the case of 4 clusters of normalized profiles of the 12641 households. The shape of average load profile of each cluster (the thick black line) shows how 4 clusters differ from each other by their consumption characteristics.

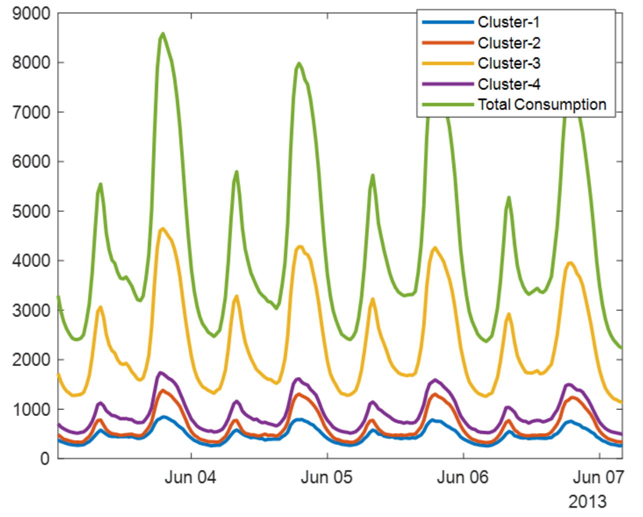

Figure 7. A sample of total consumption and clustered consumptions in case of 4-clusters.

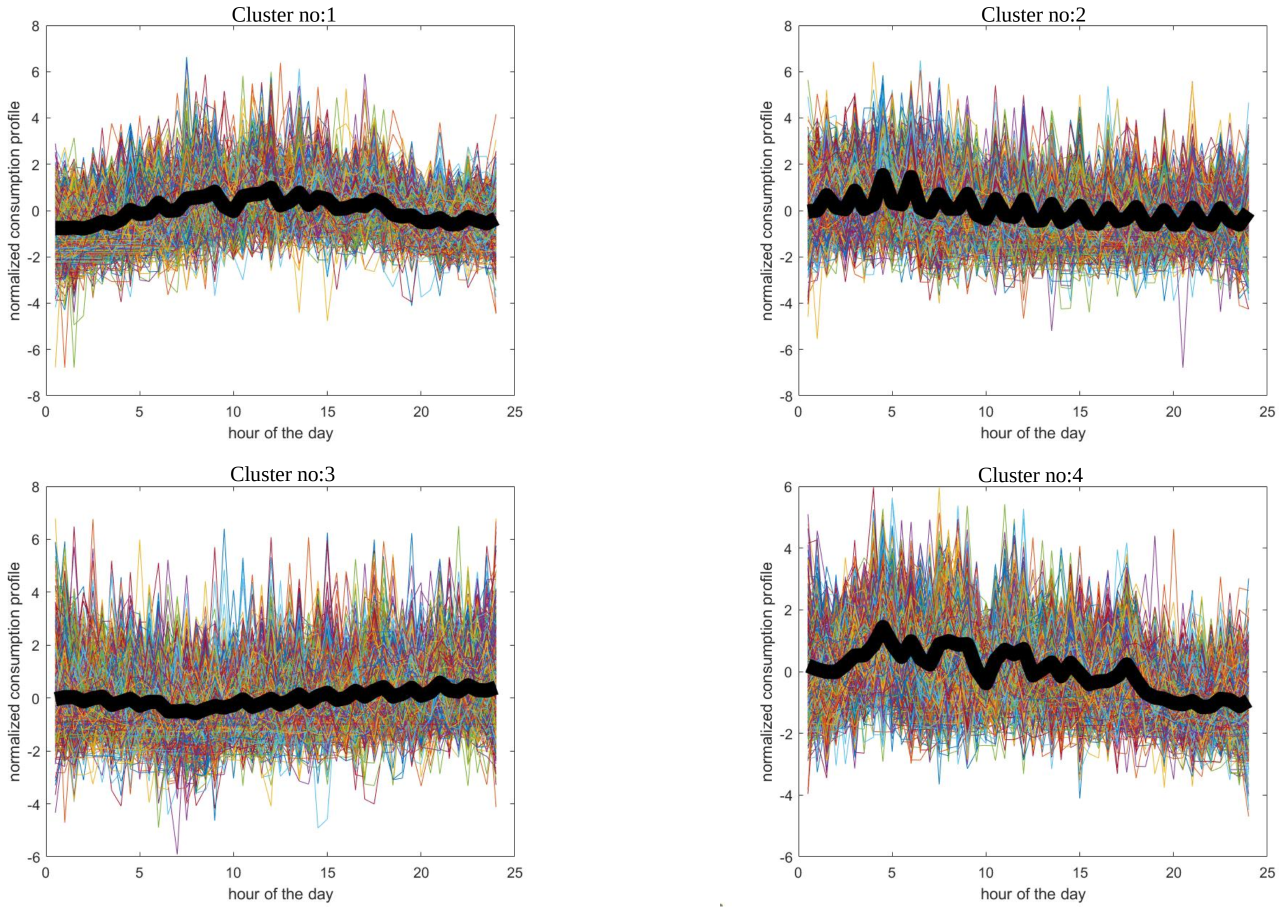

Figure 8. 4-clusters of the normalized load profiles. The tick line is the mean of the corresponding cluster.

4. RESULTS AND DISCUSSION

In this section, we compare the performance increases due to Clusters with Competing Configurations (CwCC) approach and the Clusters with the Same Configuration (CwSC) approach, with respect to the base non-aggregated (or one-cluster) forecasting performance. In evaluation of these two clustering-based approaches, 17, 5, and 36 different configurations of ARIMA, MLP, and LSTM, respectively, are used.

To be more precise in defining the two clustering-based approaches, we use the indexes in Fig. 9 to characterize the predictor and the clustering experiment. For instance, ${}^{3}_{2}P^{A}_{5}$ refers to the predictor using ARIMA method with 5th configuration trained using (and assigned to) the 2nd cluster in the experiment where the total consumption is disaggregated into 3 clusters.

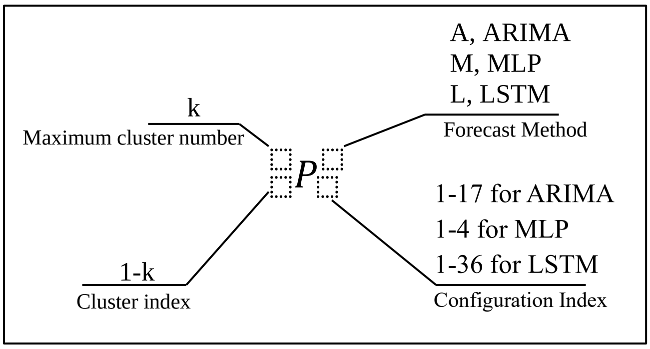

Figure 9. Indexes used in defining predictors.

Now, we explain the steps of CwCC approach using ARIMA method as an example with the help of Fig. 10. First, the (normalized) load profiles are clustered into k-clusters, where k is a predetermined number ranging from 1 to 10. Then, we train 17 differently configured ARIMA predictors, where ARIMA and the configuration space (Table-2) are predetermined, for each cluster on the total consumption data of each of them using their training part of the consumption data only. Then, for each cluster, the best ARIMA predictor is determined based on their performance on the validation part of the cluster's total consumption data. Then, to obtain the aggregate level forecasting of 12641 households, only the forecasts of the best predictors of each cluster is summed as sketched in Fig. 10.

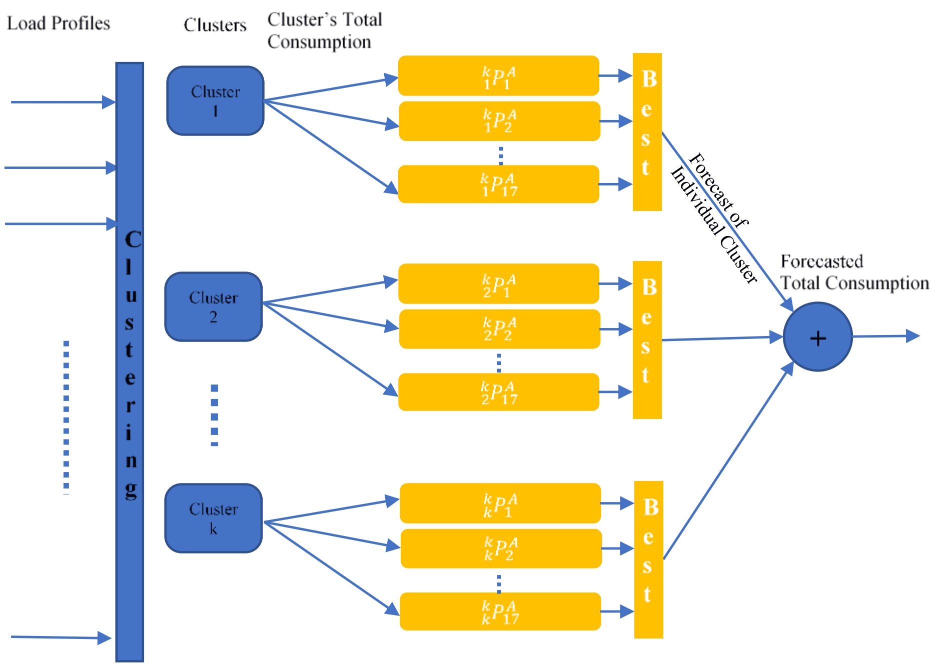

Figure 10. Flow chart for evaluating the performance of CwCC using ARIMA on forecasting of total consumption.

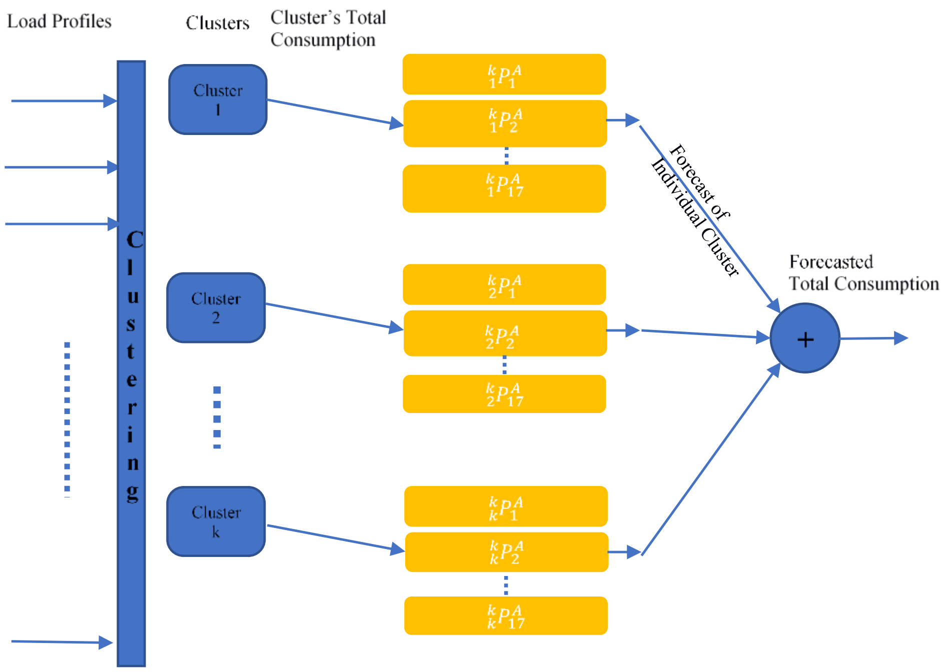

Figure 11. Flow Chart for Evaluating Performance of CwSC using 2nd configuration of ARIMA method on Forecasting of Total Consumption.

The difference of CwSC approach is that there is no best predictor selection step, thus the predictor to be used (i.e. configuration index) is predetermined (Fig. 11). For instance, in Fig. 11, the 2nd configuration of ARIMA is used in training each cluster's total consumption. Although each ARIMA model has the same configuration, their weight values differ due to training on different cluster's training data. The idea of CwSC is to picture the performance increase due to using different configurations which is best to that cluster. Since clusters are obtained using their load profiles, different lag configurations by searching for the best predictors as in CwCC is expected to give better performance. In contrast, in CwSC the same lag configurations are used across clusters.

The performances of CwSC and CwCC are reported using out-of-sample test data. That is, the test data is never introduced in the training and when picking the best predictors. In real-life usage of CwCC, an expert would have only training and validation data where he/she can design a CwCC system, whilst out-of-sample test data is where the CwCC is tested in real-life.

4.1. Performance based on ARIMA Method

We now evaluate the performances of CwSC and CwCC approaches using ARIMA as the forecasting method. In CwCC approach, each ARIMA model is trained on the train data of its corresponding cluster. Then, based on their Mean Absolute Percentage Errors (MAPE) (1) performances on the validation datasets, the best models are chosen, and assigned to that cluster. After that we arrive at a one CwCC system ready to be tested on the test dataset.

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\frac{|A_t - F_t|}{A_t} \quad (1)$$

Where n is the length of the data, t is the index, A_t is the actual value, and F_t is the forecast value.

In CwSC approach, each ARIMA model is trained on the train data of its corresponding cluster, and assigned to that cluster, and tested on the test dataset. We note that, at the end there are 17 different versions of CwSC system ready to be tested on test data, whilst there is only one CwCC forecasting system ready to be used. Thus, to report the results of these two clustering-based approaches we use boxplots or average performance for CwSC whilst we use one-line graphic for the CwCC. From one aspect, CwCC can be thought as the soft upper bound performance of CwSC approach.

The bar graphics in Fig.12a depicts the average of the train, the validation and the test performances of 17 CwSC systems in case of different number of clusters going from 1 to 10, where 1 refers to no disaggregation of 12641 household total consumption data. That is, there is only one cluster. We observe that as the number of clusters increases there are slight performance increases on the average of the train, validation and test performances. Additionally, we observe that the average train performance is the best whilst the test performance is the worst in each of the different cluster number, which is expected.

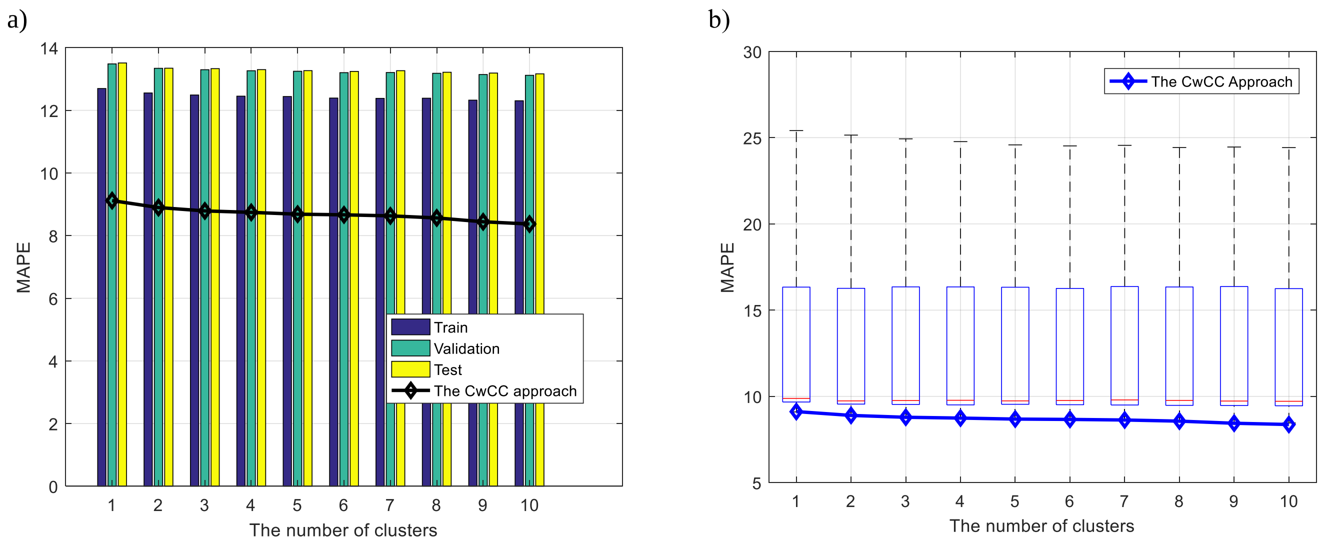

Figure 12 a) MAPE performance of the CwCC system using ARIMA method vs. CwSC system using ARIMA method. **b)** Boxplot of average performance of CwSC approach vs. the performance CwCC using ARIMA models.

The line-graphics in Fig.12a depicts the performance of CwCC system on the test data. We observe that CwCC test performance is way better than the average test performance of 17 CwSC systems (8.69 vs. 13.28 (average) respectively). To better compare the test performances of both systems, Fig. 12b depicts the boxplot (instead of average) test performances of 17 CwSC systems and the test performance of CwCC system. It is observed that CwCC system provides a minimum MAPE bound to the CwSC system.

The advantage of the CwCP approach lies in its use of the best predictor unique to that cluster. Differently than CwSC approach which uses one predictor across all clusters, there are clusters with unique predictors which outperform other predictors, and their forecasted total consumptions are summed to get final forecasted total consumption. Fig. 12b summarizes this advantage. Each cluster in the cases of different numbers of clusters (points on y-axis) has 17 ARIMA Models trained specifically to forecast that cluster's total consumption. The boxplots depict the case of using only one ARIMA model (with the same lags but different coefficients, since they are trained on different clusters). The best performances can be seen by the lower whiskers of the boxes which coincides with the CwCC approach's performance. Thus, we can conclude that CwCC approach guaranteed the best performance. In the case of CwSC approach, one ARIMA model can perform good for some clusters whilst it can perform poor for the others. For example, in the case of 10-clusters and CwCC approach, the ARIMA Model indices for each cluster are:11, 11, 17, 11, 11, 17, 13, 14, 11, 17. But in the case of CwSC approach, one configuration of ARIMA Model is used across all clusters.

4.2. Performance based on MLP Method

Our results indicate that as the number of clusters increases, there may be slight MAPE performance increases (although not monotonically) of the CwSC system on average and of the CwCC system on train, validation, and test datasets (Fig. 13a).

Figure 13b depicts the effectiveness of the CwCC system using MLP models. The performance of CwCC system is better than or equal to CwSC in 7/10 cases. And in the remaining cases, 3/10, the performance of CwCP is close to the best performance of CwSC. We note that in real life uses, the best MLP configuration of CwSC cannot be known beforehand, thus the best performance of CwSC is not guaranteed.

We observe that as the number of clusters increases there is not a monotonical increase in performance of CwCC approach using MLP models, but nevertheless the performance is better as the number of clusters increases. Thus, it is an effective clustering-based algorithm.

4.3. Performances based LSTM Method

Our results indicate that as the number of clusters increases, there may be performance increases (although not monotonically) of the CwSC approach (average) and of the CwCC approach in terms of MAPE on train, validation, and test datasets (Fig. 14a).

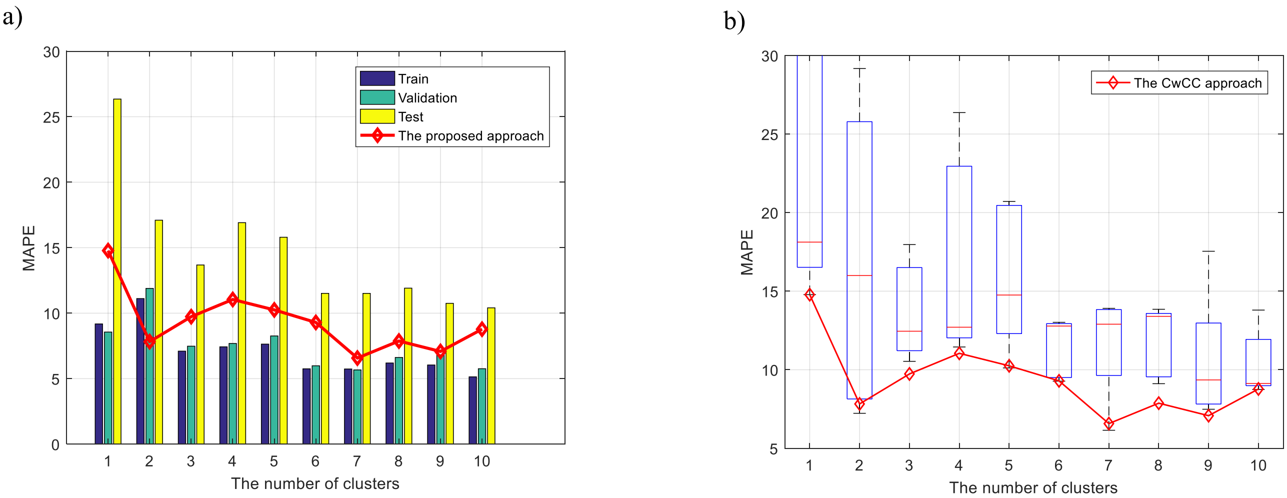

Figure 13. a) MAPE performances of the CwCC approach using MLP method vs. CwSC approach using MLP method. **b)** Boxplot of average performance of CwSC approach vs. the performance CwCC using MLP models.

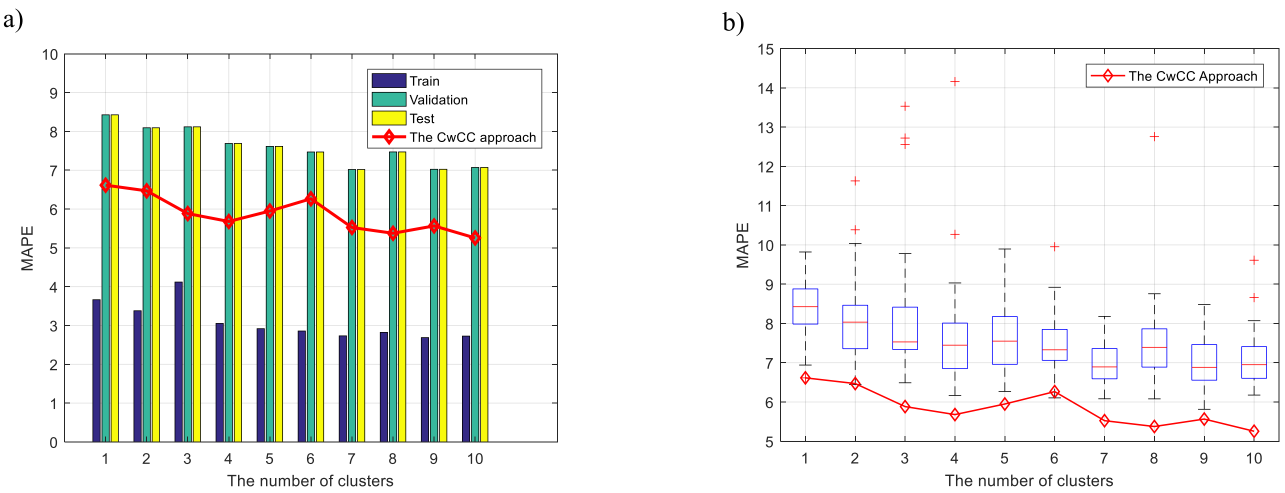

Figure 13 a) MAPE performance of CwCC approach using LSTM method vs. CwSC approach. **b)** Boxplot of average performance of CwSC approach vs. the performance CwCC using LSTM method.

4.4. Performances based LSTM Method

Our results indicate that as the number of clusters increases, there may be performance increases (although not monotonically) of the CwSC approach (average) and of the CwCC approach in terms of MAPE on train, validation, and test datasets (Fig. 14a).

Figure 14b depicts the effectiveness of the CWCC approach using LSTM models. The performance of CwCC is better than or equal to CwSC in 9/10 cases. And in the remaining case, 1/10, the performance of CwCC is close to the best performance of CwSC. Again, we note that the best LSTM configuration of CwSC cannot be known beforehand, thus the best performance of CwSC is not guaranteed. We observe that as the number of clusters increases there is not a monotonical increase in performance of CwCC approach using LSTM models, but nevertheless the performance is better than the case of non-aggregated forecasting (i.e., k=1).

Figure 14 compares the performances of CwCC approach using ARIMA, MLP, and LSTM models. The results indicate that LSTM is the best method, and it outperforms other methods, in all cases of different number of clusters.

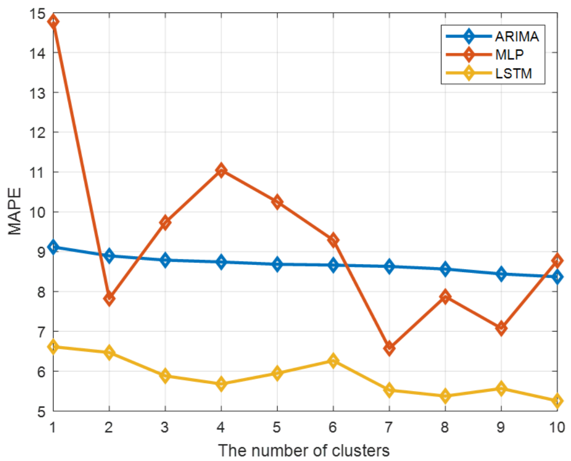

Figure 14. ARIMA, MLP, and LSTM performances vs. the number of clusters.

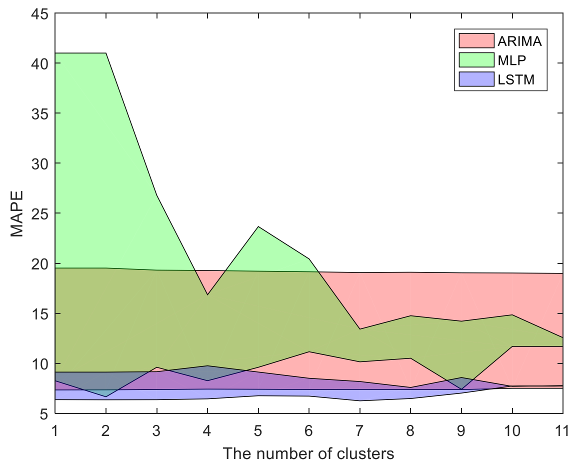

Figure 15. Mean plus standard deviations and mean minus standard deviations of ARIMA, MLP, and LSTM performances sketched as polygons vs. the number of clusters.

Moreover, Figure 15 compares the standard deviations of three methods, sketched as 2-dimensional polygons, where the upper line of each polygon is calculated as the mean plus standard deviation of the performances whilst the lower line is calculated as the mean minus standard deviation of the performances through different cluster numbers. It is observed that the performance characteristics of CwCC approach with ARIMA remain relatively stable. This might be due to the fact that ARIMA is a linear model and the sum of group forecasts would be numerically close to the aggregate-level forecast as a whole. The performance of MLP significantly improves with the number of clusters. Additionally, the standard deviation of the performances drops substantially increasing the certainty of the performance towards higher number of clusters. The performance of LSTM also increases with the number of clusters, and the standard deviation of the performances drops indicating the certainty towards higher number of clusters. As can be inferred from Figure 15, LSTM is a more reliable method to be applied in the CwCC approach than MLP especially when using a lower number of clusters. Moreover, as the number of clusters increases, the performance of LSTM statistically outperforms ARIMA and MLP in average, even surpassing their best cases. Thus, we conclude that LSTM is a statistically more reliable method to employ in a CwCC approach.

5. DISCUSSION

For clustering, we used 24-hour load profile characteristics. However, there is room for exploring novel model formulations to describe individual consumers, which can offer deeper understanding of how aggregate consumption patterns are formed. The proposed framework here can incorporate such novel formulations, as the literature in this field continues to grow.

Herein, we aimed to improve the forecast of aggregate-level individual power consumers. However, the aggregation phenomenon can appear also in other fields, such as wind forecasting, electric car availability. Thus, the proposed framework herein is likely to find application in other fields where the aggregated patterns may arise.

One drawback of our method is that it employs an ensemble of forecasters instead of a single one, which enhances the performance at the cost of increased complexity. The complexity of the method becomes a significant concern, especially when considering that demand response programs may be embedded into electronic devices rather than operating in a stand-alone computer. Additionally, in some cases the small amount of performance increase may not justify the level of complexity.

6. CONCLUSIONS AND RECOMMENDATIONS

We propose and evaluate the performance of the proposed CwCC approach for clustering-based aggregate-level short-term load forecasting using ARIMA, MLP and LSTM forecasting methods. We showed that the method is effective. Additionally, we provide clear and reproducible steps which can be useful for practicians of this field.

We believe that the subject of clustering-based load forecasting deserves more research, as there is room for improvement. The proposed CwCC approach provides one step in this direction. In the future, CwCC approach can be upgraded by adding new components and can be tested by other forecasting methods, with more clusters. Nevertheless, CwCC approach is a practical method which can be readily used in real life to increase aggregate-level forecasting performance.

Author contributions: Concept – M.A, G.D; Data Collection &/or Processing – M.A, G.D; Literature Search – M.A, G.D; Writing – M.A, G.D.
Conflict of Interest: No conflict of interest was declared by the authors.
Financial Disclosure: The authors declared that this study has received no financial support.

REFERENCES

[1] G. Pau, M. Collotta, A. Ruano, and J. Qin, "Smart Home Energy Management," Energies (Basel), vol. 10, no. 3, p. 382, Mar. 2017, doi: 10.3390/en10030382.

[2] A. Iranpour Mobarakeh, R. Sadeghi, H. Saghafi esfahani, and M. Delshad, "Techno-economic energy management of micro-grid in the presence of distributed generation sources based on demand response programs," International Journal of Electrical Power & Energy Systems, vol. 141, p. 108169, Oct. 2022, doi: 10.1016/j.ijepes.2022.108169.

[3] A. Shewale, A. Mokhade, N. Funde, and N. D. Bokde, "A Survey of Efficient Demand-Side Management Techniques for the Residential Appliance Scheduling Problem in Smart Homes," Energies (Basel), vol. 15, no. 8, p. 2863, Apr. 2022, doi: 10.3390/en15082863.

[4] A. Kahraman, O. Bulut, E. Biyik, C. Guzelis, and G. Demirkiran, "Stochastic Microgrid Control Problems: Effects of Load Distribution and Planning Horizon," in 2019 Innovations in Intelligent Systems and Applications Conference (ASYU), IEEE, Oct. 2019, pp. 1–6. doi: 10.1109/ASYU48272.2019.8946439.

[5] F. Agner, "Creating Electrical Load Profiles Through Time Series Clustering," 2019.

[6] S. Yilmaz, J. Chambers, and M. K. Patel, "Comparison of clustering approaches for domestic electricity load profile characterisation - Implications for demand side management," Energy, vol. 180, pp. 665–677, Aug. 2019, doi: 10.1016/j.energy.2019.05.124.

[7] K. Zhou, S. Yang, and Z. Shao, "Household monthly electricity consumption pattern mining: A fuzzy clustering-based model and a case study," J Clean Prod, vol. 141, pp. 900–908, Jan. 2017, doi: 10.1016/j.jclepro.2016.09.165.

[8] G. Le Ray and P. Pinson, "Online adaptive clustering algorithm for load profiling," Sustainable Energy, Grids and Networks, vol. 17, Mar. 2019, doi: 10.1016/j.segan.2018.100181.

[9] S. Lin, F. Li, E. Tian, Y. Fu, and D. Li, "Clustering load profiles for demand response applications," IEEE Trans Smart Grid, vol. 10, no. 2, pp. 1599–1607, Mar. 2019, doi: 10.1109/TSG.2017.2773573.

[10] E. Mele, C. Elias, and A. Ktena, "Electricity use profiling and forecasting at microgrid level," 2018.

[11] M. Alhussein, K. Aurangzeb, and S. I. Haider, "Hybrid CNN-LSTM Model for Short-Term Individual Household Load Forecasting," IEEE Access, vol. 8, pp. 180544–180557, 2020, doi: 10.1109/ACCESS.2020.3028281.

[12] Y. Yang, W. Li, T. A. Gulliver, and S. Li, "Bayesian Deep Learning-Based Probabilistic Load Forecasting in Smart Grids," IEEE Trans Industr Inform, vol. 16, no. 7, pp. 4703–4713, Jul. 2020, doi: 10.1109/TII.2019.2942353.

[13] C. Alzate and M. Sinn, "Improved electricity load forecasting via kernel spectral clustering of smart meters," Proceedings - IEEE International Conference on Data Mining, ICDM, pp. 943–948, 2013, doi: 10.1109/ICDM.2013.144.

[14] T. K. Wijaya, M. Vasirani, S. Humeau, and K. Aberer, "Cluster-based aggregate forecasting for residential electricity demand using smart meter data," in Proceedings - 2015 IEEE International Conference on Big Data, IEEE Big Data 2015, Institute of Electrical and Electronics Engineers Inc., Dec. 2015, pp. 879–887. doi: 10.1109/BigData.2015.7363836.

[15] A. Shahzadeh, A. Khosravi, and S. Nahavandi, "Improving load forecast accuracy by clustering consumers using smart meter data," in 2015 International Joint Conference on Neural Networks (IJCNN), IEEE, Jul. 2015, pp. 1–7. doi: 10.1109/IJCNN.2015.7280393.

[16] S. Bandyopadhyay, T. Ganu, H. Khadilkar, and V. Arya, "Individual and aggregate electrical load forecasting: One for all and all for one," in e-Energy 2015 - Proceedings of the 2015 ACM 6th International Conference on Future Energy Systems, Association for Computing Machinery, Inc, Jul. 2015, pp. 121–130. doi: 10.1145/2768510.2768539.

[17] F. Fahiman, S. M. Erfani, S. Rajasegarar, M. Palaniswami, and C. Leckie, "Improving load forecasting based on deep learning and K-shape clustering," in Proceedings of the International Joint Conference on Neural Networks, Institute of Electrical and Electronics Engineers Inc., Jun. 2017, pp. 4134–4141. doi: 10.1109/IJCNN.2017.7966378.

[18] T. Jarabek, P. Laurinec, and M. Lucka, "Energy load forecast using S2S deep neural networks with k-Shape clustering," in 2017 IEEE 14th International Scientific Conference on Informatics, IEEE, Nov. 2017, pp. 140–145. doi: 10.1109/INFORMATICS.2017.8327236.

[19] A. Cini, S. Lukovic, and C. Alippi, "Cluster-based Aggregate Load Forecasting with Deep Neural Networks," in 2020 International Joint Conference on Neural Networks (IJCNN), IEEE, Jul. 2020, pp. 1–8. doi: 10.1109/IJCNN48605.2020.9207503.

[20] 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.

[21] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” Neural Comput, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[22] Australian Government, “Smart Grid Smart City (SGSC). Customer trial data,” https://data.gov.au/dataset/ds-dga-4e21dea3-9b87-4610-94c7-15a8a77907ef/details, May 20, 2022.

[23] O. Motlagh, A. Berry, and L. O’Neil, “Clustering of residential electricity customers using load time series,” Appl Energy, vol. 237, pp. 11–24, Mar. 2019, doi: 10.1016/j.apenergy.2018.12.063.