Short-Long Correlation Based Graph Neural Networks for Residential Load Forecasting

Yiran Deng[1], Yingjie Zhou[1(✉)], and Zhiyong Zhang[2]

[1] College of Computer Science, Sichuan University, Chengdu 610065, China
{yirdeng,yjzhou09}@gmail.com
[2] Cyberspace Security Key Laboratory of Sichuan Province,
China Electronic Technology Cyber Security Co. Ltd., Chengdu 610041, China
zhiyong_zhang1013@163.com

Abstract. Accurate residential load forecasting is crucial to the future smart grid since its fundamental role in efficient energy distribution and dispatch. Compared with aggregated electricity consumption forecasting, predicting residential load of an individual user is more challenging due to the stochastic and dynamic characteristics of electricity consumption behaviors. Existing methods did not fully explore the intrinsic correlations among different types of electricity consumption behaviors, which restricts the performance of these methods. To fill this gap, this paper proposes a residential load forecasting method employing graph neural networks (GNN) to make full use of the intrinsic dependencies among various types of electricity consumption behaviors. Specifically, two kinds of graphs are constructed to leverage the dependence information, *i.e.*, short-term dynamic graphs are constructed for describing correlations among different appliances' electricity consumption behaviors only a short time ago, while long-term static graphs are built to profile a more general pattern for the internal structure of individual electricity consumption. Both short-term and long-term correlations are restricted mutually through the fusion of these two graphs. GNN is then employed to learn the implied dependencies from both the fused graphs and time-series data for load forecasting. Experiment results on a real-world dataset demonstrate the advantages of the proposed model.

Keywords: Time-series forecasting · Graph neural networks · Smart grid · Residential electricity consumption

1 Introduction

Residential load forecasting aims at obtaining an estimation of the residents' future electricity consumption. Accurate load forecasting could be used to guide the process of electricity transmission, which is able to reduce the transmission loss due to unscheduled distribution and dispatch. In smart grids, real-time load forecasting is also the fundamental step of demand response to realize peak load shaving [1]. The peak load snowballs with an increasing number of end users and devices connected to the power grids. Peak load shaving could decrease the risk of power system failure triggered by high peak load and maintain the stable operation of power grids [2]. Therefore, forecasting residential electricity consumption is of vital importance to the efficient operation of power grids [3].

Residential load forecasting could be categorized into two kinds: aggregated residential load forecasting and individual residential load forecasting. The former predicts the total load of multiple users in a region (*e.g.*, a neighborhood), while the latter estimates a single user's electricity consumption. Aggregated residential load forecasting [4–7] has been well studied. However, individual residential load forecasting is still a challenging task due to the stochasticity of various electricity consumption behaviors and the volatility of appliances' power consumption. Recently, a number of deep learning-based approaches [1,8,9] have been developed to address the problem and try to obtain satisfying performance. Nevertheless, they only utilize the overall load of a resident, neglecting the dependencies among different types of electricity consumption behaviors. Among the few methods that employ appliance-level load data to make use of the intrinsic correlations, they simply treat the correlations equally, which restricts the performance.

To overcome the limitations in existing methods, we intend to use graphs to represent the inherent correlations among electricity consumption behaviors of different appliances. Along with the historical load data reflecting the temporal correlations, they constitute a relationship that contains the aforementioned correlations, which could also be regarded as a special type of spatio-temporal correlation. We note that neither short-term nor long-term correlations could completely represent the internal correlations. On the one hand, short-term correlations contain more detailed information from recent time-series data yet introducing some unnecessary interference. On the other hand, long-term correlations reflect a general pattern of individual users' electricity consumption habits, but it lacks time-variant information. Thus, we expect to take the strengths of both short-term and long-term correlations to benefit the load forecasting.

In this paper, we propose a method based on graph neural networks (GNN) for individual residential load forecasting that fully leverages the internal correlations among various electricity consumption behaviors. To this end, two kinds of graphs—short-term dynamic graphs and long-term static graphs—are constructed to describe the short-term and long-term correlations respectively. Then, these two kinds of graphs are fused, which compensate for each other to get a more precise graph structure. Last but not least, spatio-temporal relationship learning is performed to capture spatial and temporal correlations among the fused graphs and the load data for load forecasting. Experimental results on real-world datasets show that our proposed model's forecasting performance is enhanced due to the graph construction and graph fusion strategies.

2 Related Work

In this section, we briefly introduce the related work regarding load forecasting problem. In recent years, extensive researches about aggregated residential short-term load forecasting (STLF) have been studied. Traditional aggregated residential STLF methods contain autoregressive moving average (ARMA) [4], autoregressive integrated moving average (ARIMA) [5], support vector regression (SVR) [6], *etc.*. They have been replaced by deep learning-based methods gradually, such as deep neural networks (DNN) [10], recurrent neural networks (RNN) [11] and residual neural networks (ResNet) [7].

Besides aggregated residential STLF, there are also numbers of research works conducting residential STLF for individual user [8, 9, 12]. Most of them leverage the historical information to enhance the prediction performance, while only a few works explore using the relationships among different appliances to improve the accuracy of load forecasting. Some methods considering from the aspect of appliance level achieve superior results. Razghandi *et al.* [13] forecasted the appliance-level electricity consumption of an individual house through LSTM. Dinesh *et al.* [14] proposed a method that predicts each appliance's power consumption separately using graph spectral clustering and then aggregates them to obtain the total household load forecasting. Wu *et al.* [15] put forward a multi-variate time series forecasting model based on GNN which could also be applied to individual load forecasting problems.

3 Proposed Model

In this section, we introduce the proposed residential load forecasting model *ShortLong* for individual user. An overview of the model is illustrated in Fig. 1. In general, this model is made up of three layers, 1) a graph structure extraction layer, 2) a spatio-temporal relationship learning layer and 3) an output layer. The input of the model, $X = [l_{t_1}, l_{t_2}, \dots, l_{t_P}]$, is a sequence of n appliances' load data in P time steps, where l_{t_i} represents the values of n appliances' load at time step t_i. $l_{t_i}^j$ is the load of the j^{th} appliance at time step t_i. The output of the proposed model, $Y = [l_{t_{P+1}}]$, is n appliances' load at time step $P + 1$. Details are shown as follows.

Firstly, fused graphs are obtained through the graph structure extraction layer by fusing the constructed short-term dynamic graphs and long-term static graph. Next, the model borrows the idea from the spatio-temporal graph neural network to design the spatio-temporal relationship learning layer, where interleaved temporal convolution (TC) modules and graph convolution (GC) modules capture the temporal and spatial dependencies respectively. The fused graphs are utilized as the input of the GC module. Finally, the output layer aims to project the hidden features back to one dimension to get the load forecasting of n appliances at the next time step. Residual connections are introduced to avoid gradient vanishing.

3.1 Graph Construction

Short-term Dynamic Graph Construction Short-term dynamic graph reflects correlations among different types of electricity consumption behaviors in

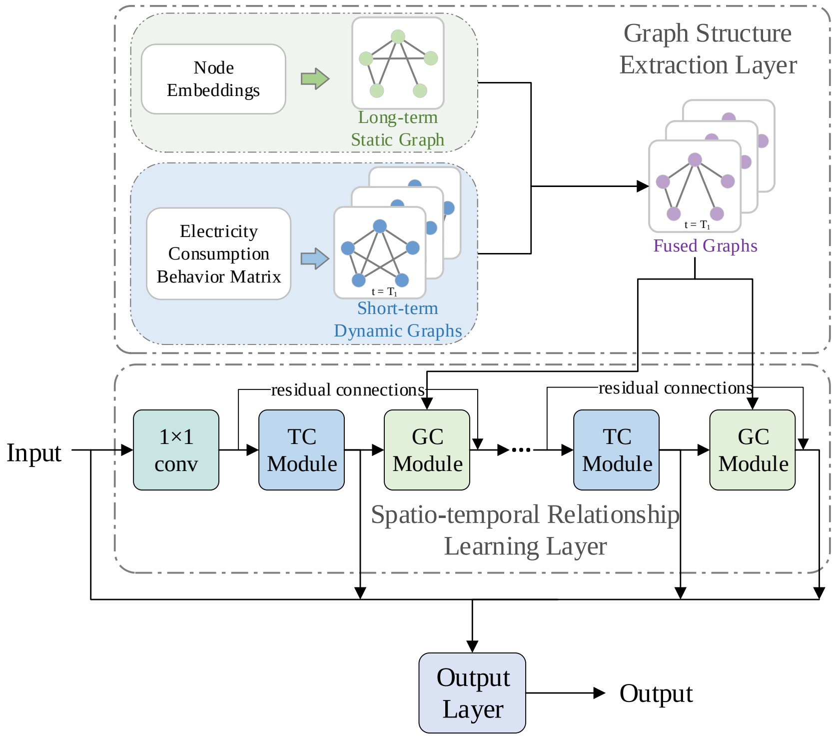

Fig. 1. The Architecture of *ShortLong* Model

a short time period. The first step to construct this kind of graph is to obtain an electricity consumption behavior matrix E regarding whether each appliance performs electricity consumption behaviors in T time intervals (a relatively short period of time). Suppose that N is the number of appliances, the size of E is $T \times N$. When an appliance is connected to a power supply at a switch-off mode or standby mode, it always yields standby power, which is small but could not be ignored. Each appliance's standby power threshold is manually chosen based on the electricity consumption behavior profile we draw. We consider that an electricity consumption behavior occurs when the actual power of an appliance is greater than the threshold, which is labeled by 1. On the contrary, 0 means there is no electricity consumption behavior happened. In this way, the original time-series data is transformed into the electricity consumption behavior matrix, which is made up of 0 and 1. E is updated every T time intervals.

Next, the short-term dynamic graph is constructed based on the electricity consumption behavior matrix E by a probability-based method. We observed that whether appliance A_i performs an electricity consumption behavior depends differentially on appliance A_j and A_k. Meanwhile, the effect of A_i consuming energy on A_j consuming energy is different from the effect of A_j consuming energy on A_i consuming energy. Consequently, the constructed dynamic graph is a directed graph with edge weights, where nodes are various appliances, and edge weights denote the correlations between appliances with respect to electricity consumption behaviors.

The short-term correlation between appliance A_i and A_j could be calculated by the following probability formula:

$$P(A_i \to A_j | t = T) = \begin{cases} \frac{N_{A_i}(T)}{N_{A_j}^{A_i}(T)} & i \neq j \\ 0 & i = j \end{cases} \tag{1}$$

where $N_{A_i}(T)$ is the number of times within T time intervals that electricity consumption behaviors of appliance A_i occurred, and $N_{A_j}^{A_i}(T)$ is the number of times within T time intervals that electricity consumption behaviors of A_j occurred after appliance A_i consuming energy. Note that a large probability value stands for a close correlation between appliances.

We could further obtain the $N \times N$ edge weight matrix W:

$$W_{ij} = \sigma(P(A_i \to A_j | t = T)) \tag{2}$$

where σ represents the sigmoid activation function.

To eliminate the accidental error, we manually choose the threshold θ to cut off some redundant edges. Then we get the optimized edge weight matrix $\widehat{W}$:

$$\widehat{W}_{ij} = \rho(W_{ij} - \theta) \tag{3}$$

where ρ represents the ReLU activation function. In this way, only essential edge relations are preserved.

Hence, the adjacent matrix A_{short} of the constructed short-term dynamic graph is defined as:

$$A_{ij}^{short} = \begin{cases} 0 & \widehat{W}_{ij} \leq 0 \\ 1 & \widehat{W}_{ij} > 0 \end{cases} \tag{4}$$

The short-term spatial correlations among appliances are represented by A_{short}, which is updated every time T intervals with the short-term dynamic graph.

Long-term Static Graph Construction Unlike the probability-based approach to construct the short-term dynamic graph, we are supposed to use a data-driven method to build the long-term static graph, which is also a directed graph concerning the correlation among appliances' usage patterns from a long time period, e.g., the whole training set.

The adjacent matrix of the long-term static graph A_{long} is obtained similar to [15]. The process to extract the long-term correlations among appliances' electricity consumption behaviors is shown as follows:

$$H_1 = \tau(\alpha \mathcal{L}(\mathbb{E}_1, \Theta_1)) \tag{5}$$

$$H_2 = \tau(\alpha \mathcal{L}(\mathbb{E}_2, \Theta_2)) \tag{6}$$

$$A_{long} = \rho(\tau(\alpha(H_1 {H_2}^T - H_2 {H_1}^T) \quad (7)$$

where τ is the tanh activation function, α determines the saturation rate of tanh function, $\mathcal{L}(\cdot, \cdot)$ is a linear layer, of which Θ_1, Θ_2 are learnable parameters, $\mathbb{E}_1$, $\mathbb{E}_2$ are node embeddings learned from the long-period data, and ρ represents the ReLU activation function. Both the subtraction term and ReLU function in Eq. 7 make A_{long} asymmetric.

3.2 Graph Fusion

Due to the dynamic characteristic of individual users' electricity consumption behaviors, the short-term dynamic graph may contain some redundant edge relations compared with the actual situation. Meanwhile, the long-term static graph suffers from the imprecise representation of the correlations for it only reflects a general usage pattern. We investigate that fusing these two graphs could make up their disadvantages and get a more comprehensive view of the spatial correlations.

The fused graph's adjacent matrix A is calculated as follows:

$$A = A_{short} \circ A_{long} \quad (8)$$

where $\circ$ represents the fusion operation. In this paper, we simply choose Hadamard product as the fusion operation.

In this way, the short-term dynamic graphs and long-term static graph are restricted mutually for the reason that graph fusion not only eliminates the redundant edge relations in short-term dynamic graphs but also preserves the necessary graph structure from a long period.

3.3 Spatio-temporal Relationship Learning for Load Forecasting

The spatio-temporal relationship learning layer consists of several interleaved GC modules and TC modules to handle the spatio-temporal dependencies of the multi-appliance time-series data. We introduce these two kinds of modules from [15]. An output layer is followed to get the final load forecasting. More details are expounded below.

Graph Convolution Module A GC module is made up of two mix-hop propagation layers, where the adjacency matrix A of the fused graph and its transpose matrix are utilized as the input respectively. The final output of GC module is the sum of the two mix-hop propagation layers' outputs.

The mix-hop propagation layer works in a two-step process, *i.e.*, the information propagation step and the information selection step. M multi-layer perceptions are applied to the second step. The output of mix-hop propagation layer P_{out} is defined as follows:

$$P_{out} = \sum_{i=0}^{M} P_i w_i \quad (9)$$

where P_i is the output of the information propagation step and w_i is the weight matrix learned by each MLP. $P_i = \beta P_{in} + (1-\beta)\widehat{A}P_{i-1}$, where β is a hyper-parameter determining the proportion of the original node states to be reserved, P_{in} is the hidden states of the previous layer, $P_0 = P_{in}$, $\widehat{A}$ is the transformed adjacency matrix of the fused graph. The calculation of $\widehat{A}$ is the same as [15].

Temporal Convolution Module A TC module contains two dilated inception layers which are inspired by dilated convolution [16]. This module aims to capture not only the potential temporal relationship but also the sequential patterns of the multi-appliance time series. The two dilated inception layers are respectively followed by a tanh activation function which is regarded as a filter as well as a sigmoid activation function which is regarded as a gate, and are gathered together in the end.

Output Layer The output layer is made up of two 2D convolution layers to project the features learned from spatio-temporal relationship learning layer to one dimension getting the final output Y:

$$Y = \mathcal{F}_1(\rho(\mathcal{F}_2(X'))) \tag{10}$$

where $\mathcal{F}_1(\cdot)$ and $\mathcal{F}_2(\cdot)$ denote two 1×1 convolution layers with different input channel size and output channel size, ρ is the ReLU activation function and X' represents the output of the spatio-temporal relationship learning layer.

4 Experiments

4.1 Experimental Setup

Dataset In this paper, we use the version 2 of the almanac of minutely power dataset (AMPds[1]) , which is commonly utilized in energy disaggregation problems. It contains 727-complete-day appliance-level load data of an individual house in Canada collected every minute ranging from April 1^{th} , 2012 to March 29^{th}, 2014. Of all the 19 appliances, we preserved 14 appliances whose electricity consumption takes up more than 1% of the total electricity usage. Besides, we convert AMPds into datasets with three kinds of time intervals, *i.e.*, 15 minutes, 30 minutes and 1 hour.

Performance Metrics To evaluate the performance of the proposed model, we choose two widely used metrics for forecasting problems, *i.e.*, mean absolute error (MAE) and mean absolute percentage error (MAPE). The definitions of them are listed below:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{11}$$

[1] https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/FIE0S4

$$MAPE = \frac{100\%}{n} \sum_{i=1}^{n} |\frac{\hat{y}_i - y_i}{y_i}| \tag{12}$$

where y_i is the actual load value and $\hat{y}_i$ is the predicted load value.

4.2 Competing Methods

ShortLong is compared with five methods, containing ARMA [4], DNN [10], LSTM [1], ResNetPlus [7] and MTGNN [15]. The chosen competing methods cover traditional statistical methods, deep-learning based methods and GNN-based methods. ARMA only use the overall load as input. MTGNN only use the appliance-level load, while the remaining models use both appliance-level load and overall load as models' inputs.

4.3 Results and Analysis

Each experiment conducted in this section is repeated for 10 times. And all the results shown in the tables and figures are the average values. The best results are highlighted. The datasets are split into a training set of 60%, a validation set of 20% and a test set of 20% in chronological order.

Main Results To evaluate the effectiveness of our proposed model, we compare *ShortLong* with five load forecasting models. The competing methods could be classified into two types, *i.e.*, methods not using graphs and GNN-based method. The first four methods of Table 1 belong to the first type, while MTGNN is a GNN-based method.

We conduct experiments on AMPds of three different time intervals. Since ResNetPlus is a model for load forecasting of one hour, we only employ ResNetPlus model on one-hour interval dataset. For our proposed model, the hyper-parameter settings of GC module and TC module are the same as those described in [15]. And the hyper-parameter T is set to 16, which will be illustrated in detail later in the parameter study section.

Table 1 shows the main results. First, we find that the results of GNN-based methods surpass those of the first type of methods on all the datasets of three time intervals, which demonstrates that utilizing graphs to represent correlations among different types of electricity consumption behaviors is useful to improve the performance. Second, *ShortLong* achieves the best results among all the competing methods. Compared with MTGNN, *ShortLong* brings a 2.90% reduction of MAPE on the 15-minute interval dataset, a 1.61% reduction of MAPE on the 30-minute interval dataset and a 4.23% reduction of MAPE on the one-hour interval dataset. The reason why *ShortLong* performs better than MTGNN is that the fusion of short-term dynamic graphs and long-term static graph provides the model with a comprehensive graph with more useful information.

Table 1. Main Results on AMPds Dataset

	interval = 15 min		interval = 30 min		interval = 1 h	
	MAE	MAPE	MAE	MAPE	MAE	MAPE
ARMA	4.1925	40.3747	10.5700	50.3961	21.4767	55.0448
DNN	3.7095	33.9846	9.6932	40.6433	21.1873	50.7147
LSTM	3.3873	30.4601	8.9172	37.3026	18.3652	40.8831
ResNetPlus	–	–	–	–	19.4258	34.9819
MTGNN	3.0994	23.1981	8.8840	26.6461	17.9697	30.1992
ShortLong	**3.0520**	**22.5258**	**8.8437**	**26.2177**	**17.6697**	**28.9231**

Ablation Study To verify that the fused graphs we construct in this paper contribute to the increase of our proposed method, we conduct the ablation study by constructing six various graphs, *i.e.*, three kinds of basic graphs and three kinds of fused graphs. Since the performance of pure data-driven approaches is not satisfying when the amount of data is relatively small, the data-driven method is not suitable to construct the short-term dynamic graphs. Thus, we construct three kinds of basic graphs, *i.e.*, long-term graph using probability-based method ($long_P$), long-term graph using data-driven method ($long_{DD}$) and short-term graph using probability-based method ($short_P$). Three fused graphs, which are respectively represented by $long_P + long_{DD}$, $long_P + short_P$ and $long_{DD} + short_P$, are constructed based on the combination of the three basic graphs. The last kind of fused graph is the one proposed in this paper. Experiments are conducted on one-hour interval AMPds dataset.

The results are shown in Table 2. In general, we observe that the performance of models with fused graphs are better than those with basic graphs. It proves that fused graphs contain more information than basic graphs in describing the internal correlations among different types of electricity consumption behaviors. Among the three models with fused graphs, the best results of MAE and MAPE are achieved when we leverage the ones fusing long-term graph using data-driven method and short-term graphs using probability-based method (*ShortLong*). In this way, we extract the graph structures from two time scales using different methods, which helps to obtain precise graph structures. Thus, the way we construct the fused graphs of *ShortLong* model is proved to be effective.

Table 2. Ablation Study on AMPds Dataset with 1-hour intervals

	MAE	MAPE
$\mathrm{long_P}$	17.7413	29.8134
$\mathrm{long_{DD}}$	17.9895	30.5641
$\mathrm{short_P}$	17.7411	30.4895
$\mathrm{long_P} + \mathrm{long_{DD}}$	17.8778	29.6328
$\mathrm{long_P} + \mathrm{short_P}$	17.6968	29.3640
$\mathrm{long_{DD}} + \mathrm{short_P}$(ShortLong)	**17.6697**	**28.9231**

Parameter Study T is the hyper-parameter determining the number of time intervals among which a short-term dynamic graph is constructed. Experiments are only conducted on AMPds dataset of one-hour time interval. We set T to 3, 12, 16, 24, 32, 64 and 84, while the other parameters remain the same as the proposed method.The line of MAPE values is shown in Fig. 2. We find that T is not sensitive when it is relatively small. Increasing T from 3 to 32 makes slight change to the MAPE value of the proposed method. However, when T becomes larger, the performance drops drastically. Among this set of experiments, the best result (MAPE=28.9231%) is achieved when $T = 16$.

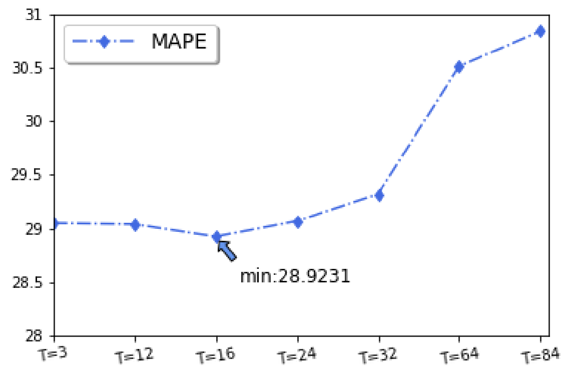

Fig. 2. Parameter Sensitivity Study on AMPds of 1-hour Interval

5 Conclusions

In this paper, we propose an effective GNN-based method *ShortLong* to address the individual residential load forecasting problem. Compared with other time series forecasting models, we fully leverage the short-term and long-term correlations among different types of electricity consumption behaviors. To be more specific, our proposed method constructs two kinds of graphs, *i.e.*, short-term dynamic graphs and long-term static graphs, from two time scales. Fused graphs with more useful information are obtained through the fusion of these two kinds of graphs. Spatio-temporal relationships for load forecasting are captured via spatio-temporal graph neural networks from the fused graphs and the load data. The performance of *ShortLong* is proved to be superior in MAE and MAPE on AMPds dataset of three time intervals compared with state-of-art methods. We consider introducing extra knowledge, such as weather conditions and public holiday information, to the graph neural networks in future works.

Acknowledgement. This work is partly supported by National Natural Science Foundation of China (NSFC) with grant number 61801315 and 62171302.

References

1. Kong, W., Dong, Z.Y., Jia, Y., Hill, D.J., Xu, Y., Zhang, Y.: Short-term residential load forecasting based on lstm recurrent neural network. IEEE Transactions on Smart Grid **10**(1), 841–851 (2017)
2. Uddin, M., Romlie, M.F., Abdullah, M.F., Abd Halim, S., Kwang, T.C., et al.: Areview on peak load shaving strategies. Renewable and Sustainable Energy Reviews **82**, 3323–3332 (2018)
3. Zhang, W., Quan, H., Srinivasan, D.: An improved quantile regression neural network for probabilistic load forecasting. IEEE Transactions on Smart Grid **10**(4), 4425–4434 (2019)
4. Huang, S.J., Shih, K.R.: Short-term load forecasting via arma model identification including non-gaussian process considerations. IEEE Transactions on power systems**18**(2), 673–679 (2003)
5. Li, W., Zhang, Z.: Based on time sequence of arima model in the application of short-term electricity load forecasting. In: 2009 International Conference on Research Challenges in Computer Science. pp. 11–14. IEEE (2009). https://doi.org/10.1109/ICRCCS.2009.12
6. Ghelardoni, L., Ghio, A., Anguita, D.: Energy load forecasting using empirical mode decomposition and support vector regression. IEEE Transactions on Smart Grid **4**(1), 549–556 (2013)
7. Chen, K., Chen, K., Wang, Q., He, Z., Hu, J., He, J.: Short-term load forecasting with deep residual networks. IEEE Transactions on Smart Grid **10**(4), 3943–3952 (2018)
8. Shi, H., Xu, M., Li, R.: Deep learning for household load forecasting—a novel pooling deep rnn. IEEE Transactions on Smart Grid **9**(5), 5271–5280 (2017)
9. Hossen, T., Nair, A.S., Chinnathambi, R.A., Ranganathan, P.: Residential load forecasting using deep neural networks (dnn). In: 2018 North American Power Symposium (NAPS). pp. 1–5 (2018). https://doi.org/10.1109/NAPS.2018.8600549
10. Sze, V., Chen, Y.H., Yang, T.J., Emer, J.S.: Efficient processing of deep neural networks: A tutorial and survey. Proceedings of the IEEE **105**(12), 2295–2329 (2017)
11. Rahman, A., Srikumar, V., Smith, A.D.: Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks. Applied energy **212**, 372–385 (2018)
12. Chaouch, M.: Clustering-based improvement of nonparametric functional time series forecasting: Application to intra-day household-level load curves. IEEE Transactions on Smart Grid **5**(1), 411–419 (2013)
13. Razghandi, M., Turgut, D.: Residential appliance-level load forecasting with deep learning. In: GLOBECOM 2020 - 2020 IEEE Global Communications Conference. pp. 1–6 (2020). https://doi.org/10.1109/GLOBECOM42002.2020.9348197
14. Dinesh, C., Makonin, S., Bajić, I.V.: Residential power forecasting using load identification and graph spectral clustering. IEEE Transactions on Circuits and Systems II: Express Briefs **66**(11), 1900–1904 (2019)
15. Wu, Z., Pan, S., Long, G., Jiang, J., Chang, X., Zhang, C.: Connecting the dots: Multivariate time series forecasting with graph neural networks. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 753–763 (2020). https://doi.org/10.1145/3394486.3403118
16. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122 (2015)