Short-Term Residential Load Forecasting based on LSTM Recurrent Neural Network

Article *in* IEEE Transactions on Smart Grid · September 2017

DOI: 10.1109/TSG.2017.2753802

Weicong Kong
UNSW Sydney

David J. Hill
The University of Hong Kong

Yan Xu
Nanyang Technological University

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSG.2017.2753802, IEEE Transactions on Smart Grid

Short-Term Residential Load Forecasting based on LSTM Recurrent Neural Network

Weicong Kong, *Student Member,* Zhao Yang Dong, *Fellow, IEEE,* Youwei Jia, *Member, IEEE,* David J. Hill, *Life Fellow, IEEE*, Yan Xu, *Member, IEEE*, Yuan Zhang, *Member, IEEE*

Abstract—**As the power system is facing a transition towards a more intelligent, flexible and interactive system with higher penetration of renewable energy generation, load forecasting, especially short-term load forecasting for individual electric customers plays an increasingly essential role in the future grid planning and operation. Other than aggregated residential load in a large scale, forecasting an electric load of a single energy user is fairly challenging due to the high volatility and uncertainty involved. In this paper, we propose a long short-term memory (LSTM) recurrent neural network (RNN) based framework, which is the latest and one of the most popular techniques of deep learning, to tackle this tricky issue. The proposed framework is tested on a publicly available set of real residential smart meter data, of which the performance is comprehensively compared to various benchmarks including the state-of-the-arts in the field of load forecasting. As a result, the proposed LSTM approach outperforms the other listed rival algorithms in the task of short-term load forecasting for individual residential households.**

Index Terms—**Short-Term Load Forecasting, Recurrent Neural Network, Deep Learning, Residential Load Forecasting**

I. Introduction

Load forecasting has been an essential task throughout the development of the modern power system. Long-term load forecasting aims to assist in power system infrastructure planning, while mid-term and short-term load forecasting can be essentially useful for system operations.

The modern power system is now moving towards a more sustainable one, whereas the increasing penetration of renewables, electric vehicles (EVs) and time-varying load demand in the distribution grids inevitably add an extra layer of system complexity and uncertainty. It is evident that the system stability is undergoing unprecedented challenges due to the intermittency of renewable generations and the complex nature of utility-customer interactions and dynamic behaviours. To this end, accurate short-term electric load forecasting on the level of residential customers can significantly facilitate the power system operations. With effective load forecasting, peak load shaving can be achieved through flexibly engaging energy storage (ES) systems or intelligent demand response (DR) technologies. From the utilities' point of view, if accurate load forecasts for individual customers are available, these electricity suppliers can rely on such information to target the best groups of customers with the highest potential to participate DR programmes in the events of power deficiency. In this case, the utilities would be more confident about the demand reduction in incentive-based DR programmes, which is important to provide load-balancing reserve and/or hedge market costs.

The ongoing worldwide expansion of smart meter infrastructure (SMI) development has laid a groundwork to drive the conventional power systems towards the smart grids. This massive deployment has also created opportunities for short-term load forecasting for individual customers.

Regarding short-term load forecasting methodologies, many approaches had been reported in the literature to address this problem. However, very few of them have confronted with individual customers directly. Customer-wise forecasts were conventionally considered trivial due to the volatile nature of individual loads. Therefore, the issues on short-term household load forecasting remain open.

Recently, *deep learning* has become one of the most active technologies in many research areas. As opposed to shallow learning, deep learning usually refers to stacking multiple layers of neural network and relying on stochastic optimisation to perform machine learning tasks. A varying number of layers can provide a different level of abstraction to improve the learning ability and task performance [1]. Especially, the long short-term memory (LSTM) recurrent neural network (RNN), which was originally introduced by Hochreiter *et al.* [2], has received enormous attention in the realm of sequence learning. Effective applications based on LSTM networks have been reported in many areas such as natural language translation [3], image captioning [4-6] and speech recognition [7]. Most of them handle data classification, but applications on regression tasks are relatively limited.

In this paper, it is aimed to make contributions to addressing the issues on short-term residential load forecasting. First, an exploratory customer-wise level data analysis is conducted to compare the difference between load on system level and loads in individual households. A density based clustering technique is employed to evaluate the inconsistency in the residential load profiles and to indicate the difficulty of the load forecasting problem for the residential sector. Justification of why LSTM is suitable for this task is given. Then, a

Manuscript was initially submitted on 13/Mar/2017, revised on 06/07/2017 and accepted on 12/09/2017.

W. Kong, Z.Y Dong and Y. Zhang are with the School of EE&T, University of New South Wales, Australia (email: weicong.kong@unsw.edu.au, joe.dong@unsw.edu.au, yuan.zhang@unsw.edu.au).

Y. Jia is with the Hong Kong Polytechnic University. (email:

D.J. Hill is with the School of EIE, University of Sydney and with the University of Hong Kong (email: dhill@eee.hku.hk).

Y. Xu is with the Nanyang Technological University, Singapore (email: eeyanxu@gmail.com).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSG.2017.2753802, IEEE Transactions on Smart Grid

residential load forecasting framework based on the LSTM is described. In addition, a conservative empirical optimisation based predictor is developed to serve as one of the many benchmarks. In the case study, we demonstrate that although the empirical predictor occasionally outperforms other machine learning based predictors in some extremely volatile cases, the LSTM can capture the subtle temporal consumption pattern persisting in single-meter load profile and produce the best forecasts for the majority of cases. On the aggregation level, at last, we also demonstrate that aggregating individual forecasts are generally more accurate than directly forecasting the aggregated loads through our proposed approach.

The rest of the paper is organised as follows. Section II provides the background of the load forecasting community. Section III conducts an explorative data analysis to visualise the challenge of the single-meter load forecasting problem. Section IV introduces the LSTM framework for short-term residential load forecasting. Section V introduces other benchmarks including two empirical predictors. The testing dataset and experimental results are given in Section VI while Section VII concludes the paper.

II. Literature Study

There have been many research works in the area of short-term load forecasting. Cao *et al.* [8] adopted autoregressive integrated moving average (ARIMA) model and similar day method for intraday load forecasting. The mechanism of their similar day method is to group the targeted day with meteorologically similar days in the history and predict the load based on the average demand of those days. It was demonstrated that in ordinary days, ARIMA performs better while similar day method wins in unordinary days. Radial basis function (RBF) neural network had been used to address the short-term load forecasting in [9, 10]. Yun *et al.* combine the RBF neural network with the adaptive neural fuzzy inference system (ANFIS) to adjust the prediction by taking account of the real-time electricity price. Li *et al.* [10] addressed the short-term day-ahead forecasting problem via a grid method combined with the back propagation (BP) and the RBF neural networks. The grid method is similar to the similar day method in [8], but instead of grouping days with similar meteorological measurement, it groups load profiles according to location, nature and size of the loads. For each group, they trained a BP network and an RBF network to predict the day-ahead load demand. Pang *et al.* [11] also proposed a neural network based predictor for very short term load forecasting. It takes the only the load values of the current and previous time steps as the input to predict the load value at the coming time step. Zhang *et al.* [12] used an ensemble of extreme learning machines (ELMs) to learn and forecast the total load of the Australian national energy market. The proposed methodology not only made use of the supreme ELM learning efficiency for self-adaptive learning but also used the ensemble structure to mitigate the instability of the forecasts. Recently, *k*-nearest neighbour (KNN) algorithm had also seen some successful examples on load forecasting [13, 14], whose dominant advantage is its efficiency. Ghofrani *et al.* proposed a dedicated input selection scheme to work with the hybrid forecasting framework using wavelet transformation and Bayesian neural network [15]. To the best of our knowledge, this approach achieved the most accurate forecasting performance on system-level load forecasting, with mean absolute percentage error (MAPE) score as small as 0.419% on average.

However, all of the above methods focus on learning and forecasting load at the system or substation level. In order to support future smart grid applications, effective load forecasting techniques for electricity users are gaining increasing interest. Zhang *et al*. [16] developed a big data architecture that combines load clustering based on smart meter data and decision tree to select corresponding load forecasting model for prediction. Quilumba *et al*. [17] also adopted the similar methodology that bases on smart meter data clustering and neural network to make intraday load forecasting at the system level. Stephen *et al.* [18] clustered and labelled daily historical data of individual households. The individual households were deemed as label sequences, which are further fit to Markov chains. Then the day ahead label can be sampled, and cluster means at each time points were used for the day ahead prediction. These works all showed that the forecasting errors could be reduced by effectively grouping different customers based on their daily profiles. However, they all only reported the aggregated load forecasting error at the system or community level where individual customer prediction errors could be offset by the diversity of different end users.

In the existing literature, Chaouch's work [19] and the work of Ghofrani *et al.* [20] are first two examples that focus on load forecasting for individual users. A functional time series forecasting approach was proposed, and the daily median absolute errors were reported in [19], but the more commonly used metric of mean absolute percentage error (MAPE) was not reported. It is thereby unsuitable to serve as a benchmark for experimental comparisons. A Kalman filter estimator was used, and an MAPE ranging from 18% - 30% was obtained in [20]. Very recently, deep learning based methods start to emerge in the load forecasting community. Ryu *et al.* showed that the load forecasting accuracy for industrial customers could be improved by using deep neural network [21]. In fact, the industrial electricity consumption patterns are much more regular than residential ones, so that a much more accurate result of about 8.85% MAPE on average is achieved compared to the results in [20]. Mocanu *et al*. employed a factored conditional restricted Boltzmann machine, one of the deep learning methods, for single-meter residential load forecasting and observed notable improvement compared to the performance of shallow artificial neural network and support vector machine [22]. Similarly, Marino *et al*. made the first attempt towards the same load forecasting issue by using LSTM [23] and demonstrate comparable results as in [22]. However, the effectiveness of the two pioneering works was only verified on the metric of root mean square error (RMSE) instead of the more common metric of MAPE, which makes it hard to contrast to other works. Also, the similar work of [23] lacked the justifiable discussion of the reason for choosing LSTM. Moreover, all the different approaches in [19, 20, 22, 23] were only verified with a single residential household, so the aggregated effect of the customer-wise load forecasts cannot be evaluated.

III. Exploratory Data Analysis and Problem Identification

The main challenge of the single household load-forecasting problem is the diversity and volatility. Thanks to the Smart Grid Smart City (SGSC) project initiated by the Australian Government [24], electrical load data for thousands of individual households are available.

A. Introduction of the Dataset

The SGSC was Australia's first commercial-scale smart grid project initiated as early as in 2010. During the 4-year project development, SGSC had gathered smart meter data for about 10,000 different customers in New South Wales. These rich data are the key to enabling further study of many smart grid technologies in a real-world context. Apparently, short-term load forecasting for individual households is one of smart grid research that can make use of this project.

Since we focus on load forecasting for a group of general individual families, it is unrealistic and time-consuming to go through every customer in the SGSC dataset. Therefore, a subset of SGSC dataset is used for the proof of concept. The selection criterion is those households which possess a hot water system, which leads to a reasonably sized testing subset including 69 customers.

B. Exploratory Data Analysis

Unlike electric load at the system level, individual residential loads lack the obvious consistent patterns that can be in favour of the forecasting accuracy. Generally, the diversity in the aggregated level smooths the daily load profiles, which make substation load relatively more predictable, while the electricity consumption of a single customer is more dependent on the underlying human behaviour.

In an individual household, the daily routines and the lifestyle of the residents and the types of possessed major appliances may have a more direct impact on the following short-term load profile. For example, some households may have somewhat fixed routines to switch on the laundry dryer after using their washing machines, which would imply the high possibility of major electricity consumption in the subsequent one or two hours.

Reference [18] also noted this phenomenon and used Practice Theory [25] to explain the root causes of energy usage on a residential level. The overall residential energy consumption is produced by the usage of appliances through an ensemble effect of 'practices', which can be interpreted as a series of doings governed by the diverse motives and intentions of people. Based on the characteristic of their underlying motives, these practices can be classified into four categories, namely 'Practical Understanding', 'Rules', 'Teleo-affective' and 'General Understanding' [18]. 'Practical Understanding' type of practices correspond to those routine activities that the actors know when to do and what to do, like taking a shower, doing laundry. 'Rules' type of practices can refer to those operations that are constrained by the technical limits of the system. For example, the dishwasher has pre-programmed washing procedure, and one can hardly control the duration and energy of the washing process. 'Teleo-affective' practices related to those goal oriented behaviours, such as making a cup of coffee and watching TV. 'General Understanding' may refer to those practices with a greater degree of persistence and regular occurrence over time, such as religion related activities. All these four types of practices from every resident in a household constitute the interaction and engagement of various electric appliances and present to be the daily electricity consumption. The diversity and complexity of human beings render the loads in different houses extremely dissimilar. Some dwellers may have more regular lifestyles so that their energy usage is easier to forecast.

In order to justify the above observation, we use a density-based clustering technique, which is known as the Density-Based Spatial Clustering of Application with Noise (DBSCAN) [26], to evaluate the consistency in daily power profile. The most obvious benefits of using DBSCAN for consistency analysis compared to other clustering techniques are that it requires no number of the cluster in the data, and it has the notion of outliers. According to the Practice Theory and [18], electricity consumption patterns will repeat with noise, so DBSCAN is an ideal clustering technique to identify outliers from a set of daily consumption profiles. If the result has fewer outliers, then the consistency is better.

In our case, each daily profile consists of 48 half-hourly readings, which can be viewed as a 48-dimension sample. DBSCAN requires three pre-determined parameters, the distance eps, the minimum number of samples in a cluster $minSam$ and the definition of the distance measure. In this paper, we set the eps equal to 10% of the average daily energy consumption, $minSam = 2$ and use the Euclidean distance as the distance measure. We apply the DBSCAN to the aggregated load and each of the 69 households over three months from 01/06/2013 to 31/08/2013, during which all 69 individual customers had complete data.

First of all, the aggregated daily profiles of all 69 households with the clustering results are shown in Fig. 1. It is clear that the daily profiles form only one cluster and have no outliers. In another word, the aggregated load is rather consistent, and its daily pattern is fairly obvious.

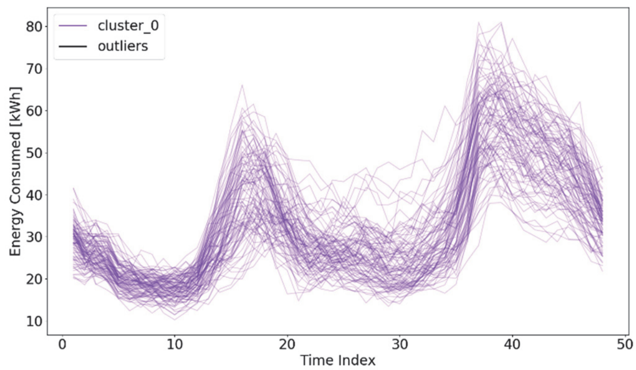

Fig. 1 Aggregated daily profiles with one cluster and no outliers

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSG.2017.2753802, IEEE Transactions on Smart Grid

For the individual customers, the number of outliers (daily profiles do not belong to any clusters) varies greatly from one household to another. This corresponds with the fact of the different lifestyles that each household may possess. The outlier distribution is shown in Fig. 2. Among all customers, we visualise the daily profiles of two opposite cases. Fig. 3 shows a case that presents fairly consistent daily profiles. There is only *one* daily profile being identified as an outlier. Apart from that, all daily profiles are grouped into two major clusters and two minor clusters. The two dominant daily patterns are visible from the figure.

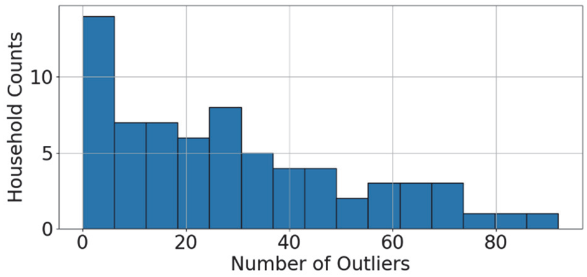

Fig. 2 The distribution of the number of outliers

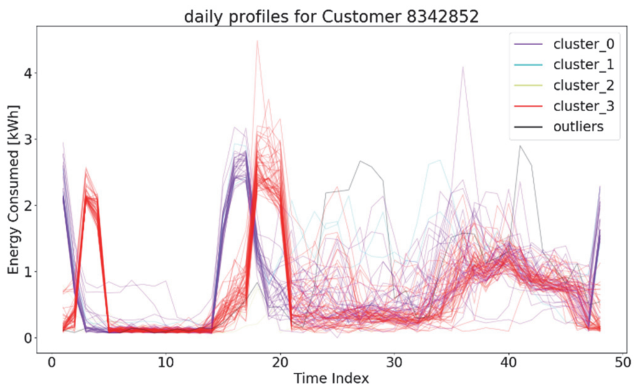

Fig. 3 A case (Customer 8342852) with only ONE outlier, two major clusters and two minor clusters

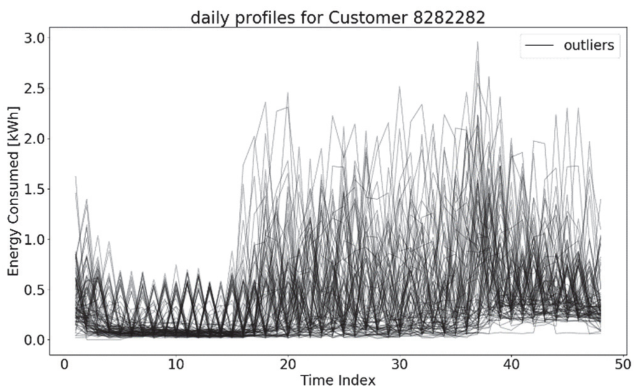

Fig. 4 A case (Customer 8282282) with no clusters at all

Fig. 4 shows a completely opposite case. This customer had no consistency at all during the 3-month period. None of the daily profiles can form even one cluster with any others. Compared Fig. 3 and Fig. 4 to Fig. 2, we can clearly see the difference between the individual loads and the aggregated load. Even in the consistent case, there exist two prominent patterns. Successful forecasting approaches that focus on the features such as time of the day and day of the week for loads on substation level may no longer be suitable for single-meter load forecasting.

In this regard, a learning algorithm with the capability of abstracting previous observations as some hidden knowledge of residents' behaviour and establishing a correlation between the abstraction and forecast target is the key to better forecasting performance. For example, as shown in Fig. 3, the algorithm should ideally learn that this customer tends to maintain energy consumption of about 2.6 kWh for about two and a half hours in the morning, regardless what the time of the day is. The LSTM recurrent network is one of the ideal candidates due to its proven capability of learning temporal correlations according to its performance in language translation and speech recognition [3, 7]. Such temporal correlations exist widely in the power consumption pattern of a single family because they are based on residents' behaviours that are hard to learn. In the case of residential load forecasting, the LSTM network is expected to be able to abstract some residents' states from the pattern of the input consumption profile, maintain the memory of the states, and finally make a prediction based on the learnt information.

IV. The Forecasting Framework Based on LSTM

A. The LSTM Model

Recurrent neural networks (RNNs) are fundamentally different from traditional feedforward neural network. They are sequence-based models, which are able to establish the temporal correlations between previous information and the current circumstances. For time series problem, this means that the decision an RNN made at time step $t-1$ could affect the decision it will reach at time step later t. Such characteristic of RNNs is ideal for the load forecasting problems of individual households, since it has been pointed out that residents' intrinsic daily routines may be one of the most important factors to the energy consumption at the later time intervals.

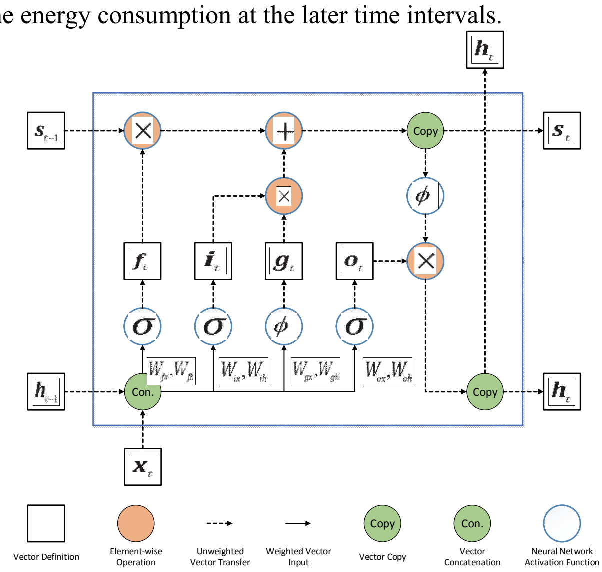

Fig. 5 The structure of an LSTM block

RNNs are trained by backpropagation through time (BPTT)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSG.2017.2753802, IEEE Transactions on Smart Grid

[27]. However, learning long-range dependencies with RNNs is difficult due to the problems of gradient vanishing or exploding [28, 29]. Gradient vanishing in RNN refers to the problems that the norm of the gradient for long-term components decreasing exponentially fast to zero, limiting the model's ability to learn long-term temporal correlations, while the gradient exploding refers to the opposite event. In order to overcome the issues, the long short-term memory (LSTM) architecture was first introduced by Hochreiter *et al.* [2] when a memory cell was included and further improved by Gers *et al.* [30] with an extra forget gate. It has been the most successful RNN architecture and received huge popularity in many subsequent applications.

Lipton *et al.* have done a detailed review of the overall structure and the latest development of LSTM [31]. To briefly introduce the concept of the LSTM, we adopt a similar naming convention as in [31]. Let $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_T\}$ denote a typical input sequence for an LSTM, where $\boldsymbol{x}_t \in \mathcal{R}^k$ represents a k-dimensional vector of real values at the tth time step. In order to establish temporal connections, the LSTM defines and maintains an internal memory cell state throughout the whole life cycle, which is the most important element of the LSTM structure. The memory cell state $\boldsymbol{s}_{t-1}$ interacts with the intermediate output $\boldsymbol{h}_{t-1}$ and the subsequent input $\boldsymbol{x}_t$ to determine which elements of the internal state vector should be updated, maintained or erased based on the outputs of the previous time step and the inputs of the present time step. In addition to the internal state, the LSTM structure also defines input node $\boldsymbol{g}_t$, input gate $\boldsymbol{i}_t$, forget gate $\boldsymbol{f}_t$ and output gate $\boldsymbol{o}_t$. The formulations of all nodes in an LSTM structure are given by (1) to (6),

$$\boldsymbol{f}_t = \sigma\left(W_{fx}\boldsymbol{x}_t + W_{fh}\boldsymbol{h}_{t-1} + \boldsymbol{b}_f\right) \quad (1)$$

$$\boldsymbol{i}_t = \sigma\left(W_{ix}\boldsymbol{x}_t + W_{ih}\boldsymbol{h}_{t-1} + \boldsymbol{b}_i\right) \quad (2)$$

$$\boldsymbol{g}_t = \phi\left(W_{gx}\boldsymbol{x}_t + W_{gh}\boldsymbol{h}_{t-1} + \boldsymbol{b}_g\right) \quad (3)$$

$$\boldsymbol{o}_t = \sigma\left(W_{ox}\boldsymbol{x}_t + W_{oh}\boldsymbol{h}_{t-1} + \boldsymbol{b}_o\right) \quad (4)$$

$$\boldsymbol{s}_t = \boldsymbol{g}_t \odot \boldsymbol{i}_t + \boldsymbol{s}_{t-1} \odot \boldsymbol{f}_t \quad (5)$$

$$\boldsymbol{h}_t = \phi(\boldsymbol{s}_t) \odot \boldsymbol{o}_t \quad (6)$$

where W_{gx}, W_{gh}, W_{ix}, W_{ih}, W_{fx}, W_{fh}, W_{ox} and W_{oh} are weight matrices for the corresponding inputs of the network activation functions; $\odot$ stands for an element-wise multiplication; σ represents the sigmoid activation function, while ϕ represents the tanh function. The LSTM block structure at a single time step is illustrated as Fig. 5, and the un-rolled LSTM architecture connecting information of the subsequent time point is shown by Fig. 6. The weights of the un-rolled LSTM are duplicated for an arbitrary number of time steps.

To train a simple one-layer LSTM recurrent network, one should specify the hyperparameter of the hidden output dimension n. In this case, the hidden output at a given time step $\boldsymbol{h}_t \in \mathcal{R}^n$, which is a n-dimensionla vector, so as to $\boldsymbol{s}_t$. The common initialization for $\boldsymbol{h}_t$ and $\boldsymbol{s}_t$ is zero initialisation, i.e. $\boldsymbol{h}_0 = \boldsymbol{0}$ and $\boldsymbol{h}_0 = \boldsymbol{0}$. There are three sigmoid functions, with output range from 0 to 1, in an LSTM block serving as the "soft" switches to decide which signals should pass the gates. If the gate is 0, then the signal is blocked by the gate. The decisions for the forget gate $\boldsymbol{f}$, the input gate $\boldsymbol{i}$ and the output gate $\boldsymbol{o}$ are all dependent on the current input $\boldsymbol{x}_t$ and the previous output $\boldsymbol{h}_{t-1}$. The signal of the input gate controls what to preserve in the internal state, while the forget gate controls what to forget from the previous state $\boldsymbol{s}_{t-1}$. With the internal state updated, the output gate decides which the internal state $\boldsymbol{s}_t$ should pass as the LSTM output $\boldsymbol{h}_t$. This process then continues to repeat for the next time step. All the weights and biases are learnt by minimising the differences between the LSTM outputs and the actual training samples. Through this un-rolled structure, information of the current time step can be stored and maintained to affect the LSTM output of the future time steps.

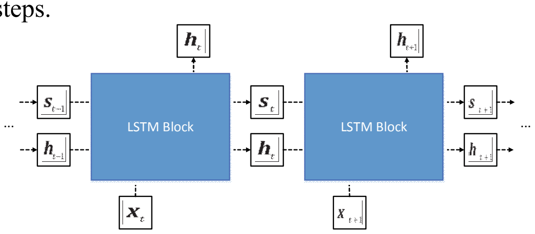

Fig. 6 The un-rolled LSTM sequential architecture

B. *The LSTM Based Forecasting Framework*

The short-term LSTM based forecasting framework is given as Fig. 7. The framework starts from data preprocessing for the inputs. In this study, the following input features are used:

1. the sequence of energy consumptions for the past K time steps $\boldsymbol{E} = \{e_{t-K}, \dots, e_{t-2}, e_{t-1}\} \in \mathbb{R}^K$;
2. the incremental sequence of the time day indices for the past K time steps $\boldsymbol{I} \in \mathbb{R}^K$, where the range for each element of $\boldsymbol{I}$ is (1 to 48);
3. the corresponding day of week indices for the past K time steps $\boldsymbol{D}$, each of which ranges from 0 to 6;
4. the corresponding binary holiday marks $\boldsymbol{H}$, each of which can either be 0 or 1.

Since the LSTM is sensitive to data scale, the four input vectors are scaled to the range of (0, 1) according to the feature's nature. We adopt the min-max normalisation for $\boldsymbol{E}$, while the vectors $\boldsymbol{I}, \boldsymbol{D}, \boldsymbol{H}$ are encoded by one hot encoder. The one hot encoder maps an original element from the categorical feature vector with M cardinality into a new vector with M elements, where only the corresponding new element is one while the rest of new elements are zeroes.

Once the four vectors are scaled to $\widetilde{\boldsymbol{E}}, \widetilde{\boldsymbol{I}}, \widetilde{\boldsymbol{D}}, \widetilde{\boldsymbol{H}}$, the input for LSTM layer is a matrix of the concatenation of the four, i.e.:

$$\boldsymbol{X} = \left\{\widetilde{\boldsymbol{E}}^{\mathrm{T}}, \widetilde{\boldsymbol{I}}^{\mathrm{T}}, \widetilde{\boldsymbol{D}}^{\mathrm{T}}, \widetilde{\boldsymbol{H}}^{\mathrm{T}}\right\}$$

Each row of the input matrix is the scaled features for the corresponding time step, which is fed to corresponding LSTM block in the LSTM layer. Due to the sequential nature of the output of an LSTM layer, an arbitrary number of LSTM layers can be stacked to form a deep learning network.

The outputs of the top LSTM layer are fed to a conventional feedforward neural network which maps the intermediate LSTM outputs to a single value, i.e. the energy consumption forecast of the target time interval.

All frameworks for all customers are built on a desktop PC with a 3.4 GHz Intel i7 processor and 8GB of memory using the Keras library [32] with Theano backend [33]. After run-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSG.2017.2753802, IEEE Transactions on Smart Grid

ning some initial tests, the computationally efficient Adam optimiser [34] showed slightly better results than other candidates, including the classic stochastic gradient descent (SGD), Adagrad [35], Adadelta [36] and RMSProp [37]. Therefore, Adam is used for the training of the proposed forecasting framework with recommended default parameters (learning rate, momentum and decay).

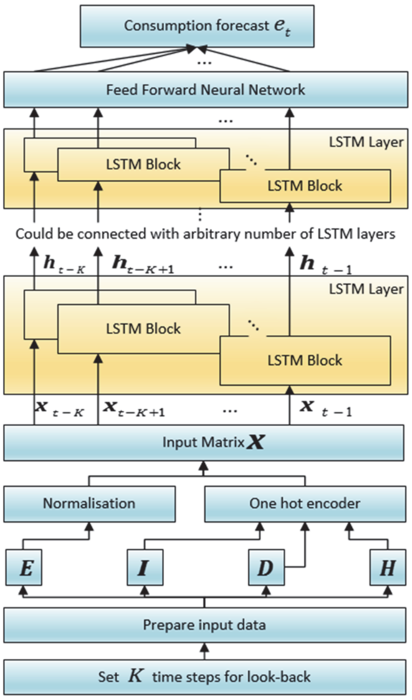

Fig. 7 The LSTM based forecasting framework

V. THE BENCHMARKS

Due to the highly volatile nature of single-user power consumption, the forecasting accuracy is usually low in the existing literature despite the state-of-the-art approaches were applied. Therefore, in addition to some machine learning methods, we also investigate some empirical methods to serve as the benchmarks in the present work.

A. Empirical Approaches

1) Empirical Means

We start with one of the most naïve predictors – the empirical mean predictor. It simply outputs a forecasting value based on the statistical mean given the customer ID, the time of the day and the day type (weekday or weekend).

2) Empirical MAPE Minimisation

We further develop a more sophisticated optimisation based empirical predictor to enrich the benchmarks. Similar to the empirical mean predictor, the empirical MAPE minimisation predictor is based on statistical energy consumption distribution, resembling the models used in [38]. For example, given the customer ID, the time of the day, the day type and the desired precision level (e.g. one watt-hour), an empirical probability mass function (pmf) can be derived from historical data, as shown by the histogram in Fig. 8. With the empirical pmf ready, the MAPE expectation of any forecasts can be calculated. Let p denote an arbitrary predicted load value within the empirical range and p_i denote the ith possible discretised value also within the empirical range, the MAPE expectation of choosing arbitrary predicted load value p as the forecast is

$$\mathcal{E}_{id,t,h}(p) = \sum_i pmf_{id,t,h}(p_i)\left|\frac{(p-p_i)}{p_i}\right| \tag{7}$$

where $pmf_{id,t,h}()$ is the empirical probability mass function derived by specifying the customer id, the time of day t and the day type h. By minimising the MAPE expectation, load forecast $\hat{p}$ at time t when the day type is h can be obtained as:

$$\hat{p}_t = \operatorname*{argmin}_p\left(\mathcal{E}_{id,t,h}(p)\right) \tag{8}$$

The empirical MAPE minimization forecast for Customer 8804804 at 12:00 in a weekday is illustrated in Fig. 8. The red line indicates the MAPE expectation throughout the empirical range for this customer in the interval. The minimal MAPE expectation point is more conservative than the empirical mean of the data.

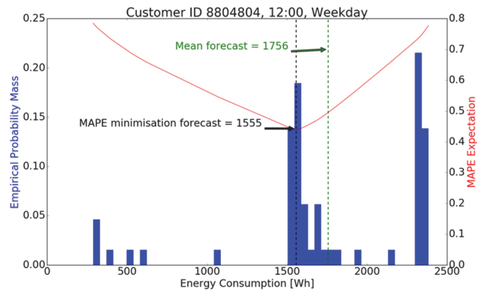

Fig. 8 The MAPE minimisation forecast and the empirical mean forecast

B. Some State-of-the-art Machine Learning Approaches

Due to many promising cases of neural network based short-term load forecasting on the grid level [9-11], we should also benchmark the results of our proposed framework against those from the conventional backpropagation neural network (BPNN). Two scenarios of BPNN are considered. One of the scenarios only features the energy consumption of the corresponding consumption in the past D days, which is denoted as BPNN-D hereafter. The other looks back the energy consumption of the past K time steps as our proposed LSTM based framework, which is denoted as BPNN-T.

Reasonable performances based on the k-nearest neighbour (KNN) regression [13, 14], superior forecasting accuracy based on extreme learning machine (ELM) [12] and the most accurate results from a sophisticated input selection scheme combined with a hybrid forecasting framework (IS-HF) [15] had been demonstrated. Therefore, they are also included as our benchmark candidates.

VI. CASE STUDY

A. The Test Settings

Due to the incompleteness of the data, we pick a 3-month period from 01-Jun-2013 to 31-Aug-2013, when all selected customers have complete data 30-minute interval energy consumption data. This period also covers the whole winter season of New South Walse, Australia, so that seasonal factor can

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSG.2017.2753802, IEEE Transactions on Smart Grid

be reasonably ignored in this case study.

The period spans over 92 days. The data are split into three subsets for different purposes, namely training set (from 01-Jun-2013 to 05-Aug-2013), validating set (from 06-Aug-2013 to 22-Aug-2013) and testing set (from 23-Aug-2013 to 31-Aug-2013). In other words, the data split is 0.7/0.2/0.1. The training set is used to train forecasting models, and the validating set is used to select best performing models, while the testing set is used for result evaluation at last. In this case, there are nine days in total for evaluation. Considering the SGSC data are with 30-minute intervals, there are totally 29,808 time points to forecast for all customers in the group. For the empirical approaches, both the training and validating sets are combined to derive the statistics for each user.

In general, hyper-parameter tuning is a big topic and is essential to obtain the best forecasting performance. However, since we are forecasting individually, tuning 69 models for each of the candidate methods is very time-consuming for this proof-of-concept paper. In this work, we only focus on identifying the candidate method(s) with the best overall performance. Therefore, some rules of thumb for hyperparameter selection is adopted. For deep learning models, previous work suggests that the performance of the networks is relatively insensitive to any combination of some layers and layer size [39]. This is also confirmed in the paper [23] that also adopts the LSTM technique. According to the consistent finding in [40], multiple layers always work better than a single layer, and the number of hidden nodes should be sufficiently large. In this case, we connect both the LSTM and the BPNN with two hidden layers and 20 hidden nodes on each hidden layer to allow horizontal comparisons. The number of neighbours for the KNN algorithm is set to 20 with uniform weight scheme. For the parameter setting of ELM, it is typically a single layer feedforward network. Due to ELM's advantageous training efficiency, an exhaustive search strategy is employed to get the optimal number of hidden neurones. For the IS-HF, we strictly follow the structure and parameters introduced in [15]. For the neural network parameters in IS-HF, we use one hidden layer with six neurons. Parameters of all models are summarised in Table I. The total number of epochs for both LSTM and BPNN is set to 150. Due to the nature of random initialisation of ELM, there is no concept of an epoch in the original ELM framework. Therefore, the training of ELM repeats 150 times to find the best model to perform forecasting.

TABLE I HYPER-PARAMETER SUMMARY

Method	*# hidden layers*	*# hidden nodes/layer*	*# Epochs*
LSTM	2	20	150
BPNN	2	20	150
KNN	n/a	20 neighbours	n/a
ELM	1	Varied	150*
IS-HF	1	6	n/a

B. Experimental Results and Discussion

Multiple scenarios with different time horizons of {2, 6, 12} look-back time steps, for each machine learning method are tested. For the BPNN-D, scenarios up to 3 days in the past are studied. For each scenario, the mean absolute percentage error (MAPE) is calculated for test time intervals of all customers. The testing forecasting MAPEs are given in Table II.

1) Performance for individual households

The second column of Table II shows the comparison of the average MAPE of all 29,808 individual forecasts. Generally, the forecasts for each time interval of each household are not as accurate as the forecasts for substation loads. This level of accuracy is anticipated because it was reported that the forecasting accuracy tends to drop significantly as the level of aggregation decreases [41], from 1.97% MAPE at the national level and 5.15% at university campus level to 13.8% at the village level. Reference [42] also provides a range of MAPE between 10% - 35% for a specific case of a residential feeder. In our case, as we attempt to forecast the most granular level of loads, the accuracies achieved in Table II for various methods are reasonable.

The results can be divided into three tiers according to the range of the average MAPE, namely below 50%, below 100% and above. For the top tier, only three methods achieve the level of accuracy of MAPE below 50%. They are our proposed LSTM forecasting framework, the proposed statistics-based empirical MAPE minimisation predictor and the BPNN considering the consumption of the time interval prior to the target time step. The LSTM framework is generally the best, followed by the MAPE minimisation predictor. The best score is 44.06% achieved by the proposed LSTM method with 12 look-back steps. The average of the three LSTM scenarios is 44.25%, which is generally better than the BNPP-T approaches (with an average of the three scenarios 49.38%) in the same tier by about five percentage points.

KNN, BPNN-D and IS-HF fall in the second tier of forecasting performance. The KNN achieves MAPE from 71% to 81% by also using inputs of the energy consumption from the previous time intervals. The BPNN using the input of energy consumption from the same time intervals of previous days achieves MAPE from 74% to 80%. It is comparative to the KNN approaches. The IS-HF, which is reported to achieve supreme accuracy on a system level, has only 96.76% of MAPE for the individual customers in this test case. The performance gap between the top tier and the second is obvious.

ELM, as another state-of-the-art machine learning approach that achieves superior accuracy on a system level, completely fails for the tasks of single-meter load forecasting. Its MAPE range from 122% to 136%, which is only marginally better than the most naïve approach – the empirical mean predictor.

TABLE II LOAD FORECASTING MAPE SUMMARY

Method/Scenario	*Avg. MAPE individual forecasts*	*Avg. MAPE Aggregating forecasts*	*Avg. MAPE forecasting the aggregate*
LSTM/2 time steps	**44.39 %**	**8.18%**	**9.14%**
LSTM/6 time steps	**44.31%**	**8.39%**	**8.95%**
LSTM/12 time steps	**44.06%**	**8.64%**	**8.58%**
Empirical mean	136.46%	32.54%	32.54%
MAPE minimisation	**46.00%**	34.91%	27.28%
BPNN-D/1 day	80.02%	11.69%	14.50%
BPNN-D/2 days	75.28%	11.67%	14.48%
BPNN-D/3 days	74.10%	11.66%	14.42%
BPNN-T/2 time steps	49.62%	**8.37%**	9.54%
BPNN-T/6 time steps	49.04%	**8.29%**	9.55%
BPNN-T/12 time steps	49.49%	**8.36%**	9.17%
KNN/2 time steps	74.83%	15.37%	11.23%
KNN/6 time steps	71.19%	14.61%	12.10%

KNN/12 time steps	81.13%	15.23%	15.30%
ELM/2 time steps	122.90%	33.68%	Not tested
ELM/6 time steps	136.49%	35.35%	Not tested
ELM/12 time steps	123.45%	30.05%	Not tested
IS-HF	96.76%	20.43%	32.09%

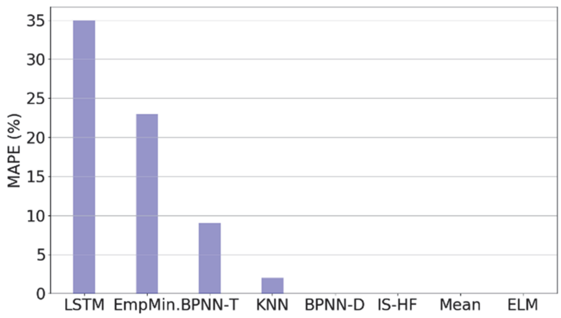

Fig. 9 The Statistics of the best approaches

The statistics of the best methods for all 69 households are shown in Fig. 9. The LSTM performs the best for 35 out of the 69 households, which accounts for 50% of the test subjects. The empirical MAPE minimisation approach is the best predictor for 23 customers, while the BPNN-T and KNN are the best for 9 and 2 households respectively. The rest of the benchmarks do not lead the forecasting performance on for any households.

2) Performance for aggregated loads

Instead of giving forecast evaluation on individual customers, many research works only report the forecasting error on the aggregated level [16-18]. That is to aggregate all the forecasts for individual customers and compare them to the aggregated actual loads. As shown in the third column of Table II, all the results correspond to the range of residential forecasting MAPE of 10% - 35% suggested by [42]. The LSTM and the BPNN-T in the top tier are still in the dominant position, outperforming all the other benchmarks by noticeable margins with MAPE range from 8.18% to 8.64%. 69 customers can form the size of a village effectively. By aggregating the individual forecasts from the proposed LSTM framework, we achieve much better forecasts compared to the MAPE scores of 13.8% [41] and 11.13% (123 households) [18] on the same aggregation level. The MAPE minimisation approach, as one of the top-tier predictors for individual load forecasting, becomes the worst in the aggregated scenario.

3) Aggregating forecasts versus forecasting the aggregate

In addition to aggregating the forecasts, the aggregated load forecasting can also be carried out by the conventional strategy of forecasting the aggregated load directly. We apply all the top- and the second-tier approaches and compare their results in the fourth column of Table II.

It should be noted that the performances of empirical mean by either way are identical, but other forecasting approaches demonstrate different accuracy under different strategies. The strategy of aggregating the forecasts generally provides more accurate forecasting performance than forecasting the aggregate for the two top-tier approaches. Aggregating the forecasts for both LSTM and BPNN-T sees about 0.49-percentage-point and 1.08-percentage-point improvement respectively.

In addition, for the LSTM and the BPNN-T approaches in the first tier, the LSTM once again outperforms the BPNN-T by directly forecasting the aggregated loads. However, due to the difference between of load characteristics on the aggregation level and the individual residential level, the advantage of LSTM performance on aggregated loads is weakened. Recall that the LSTM is designed to track and learn the temporal relationships of energy consumption. Such relationship may stem from human's daily routine practices such as the patterns shown in Fig. 3. When this kind of patterns is not so prominent in the load on aggregation level, the advantage of using LSTM is not as large as applying it on individual loads.

In summary, the LSTM and the BPNN-T approaches are the two best methods for the tasks of individual residential load forecasting. More specifically, LSTM has much better performances for single-meter load forecasting than BPNN-T due to its capability of establishing a temporal relationship and the load nature on the granular level. The LSTM performance edge on aggregation level is not as significant as on the granular level. However, load forecasting on aggregation level can be further improved by aggregating the granular forecasts.

C. Tracking the Internal State of the LSTM

Recall that the customer 8342852 as shown in Fig. 3 demonstrates two major daily consumption patterns. For this particular customer, the LSTM achieves the MAPE of 36.52%, while the second-best BPNN-T only gets 58.97%. The comparison is shown in the first row of Fig. 10. It can be seen that the LSTM forecast follows the peaks much better than the BPNN-T. The BPNN-T always forecasts peaks after the peaks had already happened.

In order to better show how the LSTM forecasts load at this residential level, we also visualise all the internal states of the LSTM in the middle row of Fig. 10. Each curve represents the dynamic tracking of the internal state for a hidden node in the network. The summation of all internal states is shown as the third row of Fig. 10. From the summation of the LSTM states, we can visualise five cycles, which correspond to the five-day period (240 time steps). The internal states of the LSTM are the key for the algorithm to develop a long-term temporal relationship and help it outperform the BPNN.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSG.2017.2753802, IEEE Transactions on Smart Grid

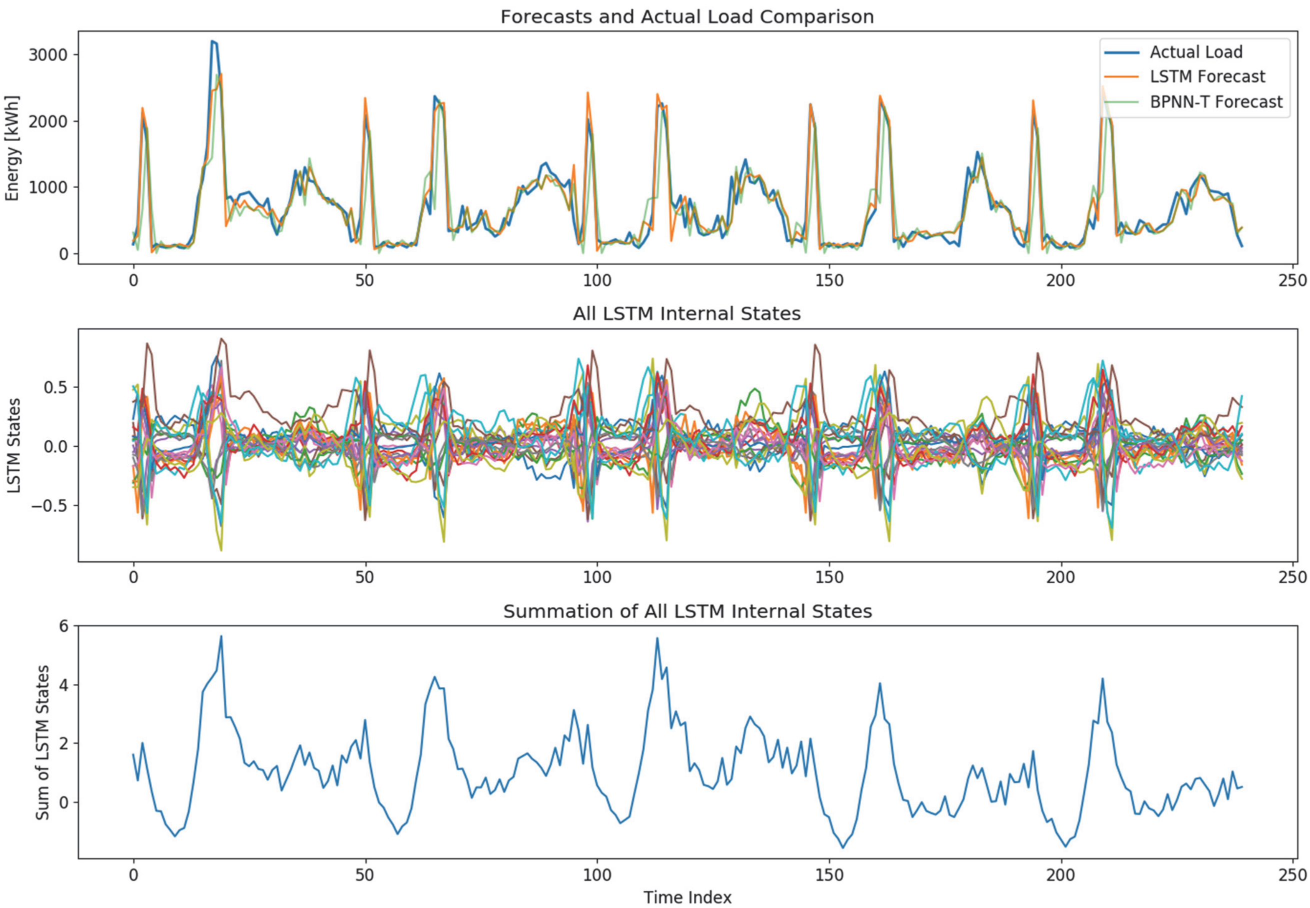

Fig. 10 Forecast comparison and LSTM internal state visualisation

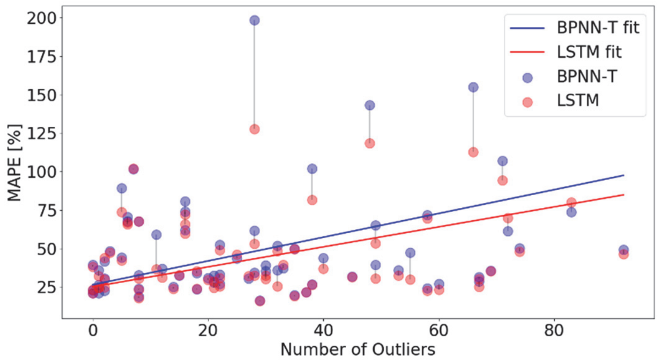

Fig. 11 MAPE versus number of outliers for LSTM and BPNN-T

D. The Load Consistency and the Forecasting Accuracy

Every household has different life consistency, which lead to different difficulty of the forecasting task. In section III, we had introduced the DBSCAN clustering algorithm and used the number of outliers as an indicator of the lifestyle inconsistency of each household. Fig. 11 shows both the results of LSTM and BPNN-T for all customers. The blue dots represent the results achieved by the BPNN-T, while the red dots represent the MPAE of LSTM. Two linear regression curves are fit to each group of the results. It turns out that as the inconsistency of lifestyle grows, the energy consumption becomes more difficult to forecast for both approaches. The LSTM approach generally improves the forecasting performance. However, when the inconsistency is relatively small, the improvement is also marginal; the improvement of LSTM tends to be more obvious as the inconsistency gets larger.

VII. CONCLUSION

This paper tries to address the short-term load forecasting problem for individual residential households. First, the difference between electric load on such granular level and load on substation level is elaborated. Unlike the load on system or substation level where daily consumption patterns always exist, energy consumption in a single household is usually volatile. A density based clustering technique is employed to evaluate and compare the inconsistency between the aggregated load and individual loads. According to the Practice Theory, lifestyles of the residents will be reflected in the energy consumption as the repetitive patterns no matter how inconsistent it is. Therefore, this paper proposes an LSTM recurrent neural network based load forecasting framework for this extremely challenging task of individual residential load forecasting, because LSTM has been proven to learn the long-term temporal connections.

Multiple benchmarks are comprehensively tested and compared to the proposed LSTM load forecasting framework on a real-world dataset. It turns out that many load forecasting approaches which are successful for grid or substation load forecasting struggle in the single-meter load forecasting problems. The proposed LSTM framework achieves generally the best forecasting performance in the dataset.

Moreover, although individual load forecasting is far from accurate, aggregating all individual forecasts yields better forecast for the aggregation level, compared to the conventional strategy of directly forecasting the aggregated load.

The inconsistency in daily consumption profiles generally affects the predictability of the customers. The higher the inconsistency is, the more the LSTM can contribute to the forecasting improvement compared to the simple back propagation neural network.

As for future work, methodologies for parameter tunning can be developed to further enhance forecasting accuracy on different types of customers, especially those households with very high volatility.

REFERENCES

[1] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 35, pp. 1798-1828, 2013.

[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation,* vol. 9, pp. 1735-1780, 1997.

[3] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104-3112.

[4] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3156-3164.

[5] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3128-3137.

[6] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille, "Deep captioning with multimodal recurrent neural networks (m-RNN)," *arXiv preprint arXiv:1412.6632,* 2014.

[7] A. Graves and N. Jaitly, "Towards End-To-End Speech Recognition with Recurrent Neural Networks," in *ICML*, 2014, pp. 1764-1772.

[8] X. Cao, S. Dong, Z. Wu, and Y. Jing, "A Data-Driven Hybrid Optimization Model for Short-Term Residential Load Forecasting," in *Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), 2015 IEEE International Conference on*, 2015, pp. 283-287.

[9] Z. Yun, Z. Quan, S. Caixin, L. Shaolan, L. Yuming, and S. Yang, "RBF Neural Network and ANFIS-Based Short-Term Load Forecasting Approach in Real-Time Price Environment," *IEEE Transactions on Power Systems,* vol. 23, pp. 853-858, 2008.

[10] H. Li, Y. Zhao, Z. Zhang, and X. Hu, "Short-term load forecasting based on the grid method and the time series fuzzy load forecasting method," in *International Conference on Renewable Power Generation (RPG 2015)*, 2015, pp. 1-6.

[11] P. Qingle and Z. Min, "Very Short-Term Load Forecasting Based on Neural Network and Rough Set," in *Intelligent Computation Technology and Automation (ICICTA), 2010 International Conference on*, 2010, pp. 1132-1135.

[12] R. Zhang, Z. Y. Dong, Y. Xu, K. Meng, and K. P. Wong, "Short-term load forecasting of Australian National Electricity Market by an ensemble model of extreme learning machine," *IET Generation, Transmission & Distribution,* vol. 7, pp. 391-397, 2013.

[13] R. Zhang, Y. Xu, Z. Y. Dong, and W. Kong, "A Composite k-Nearest Neighbor Model for Day-Ahead Load Forecasting with Limited Temperature Forecasts," presented at the IEEE General Meeting Boston, 2016.

[14] F. H. Al-Qahtani and S. F. Crone, "Multivariate k-nearest neighbour regression for time series data - A novel algorithm for forecasting UK electricity demand," in *Neural Networks (IJCNN), The 2013 International Joint Conference on*, 2013, pp. 1-8.

[15] M. Ghofrani, M. Ghayekhloo, A. Arabali, and A. Ghayekhloo, "A hybrid short-term load forecasting with a new input selection framework," *Energy,* vol. 81, pp. 777-786, 3/1/ 2015.

[16] P. Zhang, X. Wu, X. Wang, and S. Bi, "Short-term load forecasting based on big data technologies," *CSEE Journal of Power and Energy Systems,* vol. 1, pp. 59-67, 2015.

[17] 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 Transactions on Smart Grid,* vol. 6, pp. 911-918, 2015.

[18] B. Stephen, X. Tang, P. R. Harvey, S. Galloway, and K. I. Jennett, "Incorporating Practice Theory in Sub-Profile Models for Short Term Aggregated Residential Load Forecasting," *IEEE Transactions on Smart Grid,* vol. PP, pp. 1-8, 2016.

[19] M. Chaouch, "Clustering-Based Improvement of Nonparametric Functional Time Series Forecasting: Application to Intra-Day Household-Level Load Curves," *IEEE Transactions on Smart Grid,* vol. 5, pp. 411-419, 2014.

[20] M. Ghofrani, M. Hassanzadeh, M. Etezadi-Amoli, and M. S. Fadali, "Smart meter based short-term load forecasting for residential customers," in *North American Power Symposium (NAPS)*, 2011, pp. 1-5.

[21] S. Ryu, J. Noh, and H. Kim, "Deep neural network based demand side short term load forecasting," in *2016 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2016, pp. 308-313.

[22] E. Mocanu, P. H. Nguyen, M. Gibescu, and W. L. Kling, "Deep learning for estimating building energy consumption," *Sustainable Energy, Grids and Networks,* vol. 6, pp. 91-99, 6// 2016.

[23] D. L. Marino, K. Amarasinghe, and M. Manic, "Building energy load forecasting using Deep Neural Networks," in *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, 2016, pp. 7046-7051.

[24] Smart Grid, Smart City [Online]. Available: http://www.industry.gov.au/ENERGY/PROGRAMMES/SMARTGRID SMARTCITY/Pages/default.aspx

[25] E. Shove, M. Pantzar, and M. Watson, *The dynamics of social practice: Everyday life and how it changes*: Sage, 2012.

[26] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Kdd*, 1996, pp. 226-231.

[27] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE,* vol. 78, pp. 1550-1560, 1990.

[28] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks,* vol. 5, pp. 157-166, 1994.

[29] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," ed: A field guide to dynamical recurrent neural networks. IEEE Press, 2001.

[30] F. A. Gers, r. A. Schmidhuber, and F. A. Cummins, "Learning to Forget: Continual Prediction with LSTM," *Neural Comput.,* vol. 12, pp. 2451-2471, 2000.

[31] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv:1506.00019,* 2015.

[32] F. Chollet. (2015). *Keras*. Available: https://github.com/fchollet/keras

[33] R. Al-Rfou, G. Alain, A. Almahairi, C. Angermueller, D. Bahdanau, N. Ballas, *et al.*, "Theano: A Python framework for fast computation of mathematical expressions," *arXiv preprint arXiv:1605.02688,* 2016.

[34] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," presented at the 3rd International Conference for Learning Representations, San Diego, 2015.
[35] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research,* vol. 12, pp. 2121-2159, 2011.
[36] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701,* 2012.
[37] G. Hinton, N. Srivastava, and K. Swersky, "RMSProp: Divide the gradient by a running average of its recent magnitude," *Neural networks for machine learning, Coursera lecture 6e,* 2012.
[38] S. M. Mousavi and H. A. Abyaneh, "Effect of Load Models on Probabilistic Characterization of Aggregated Load Patterns," *IEEE Transactions on Power Systems,* vol. 26, pp. 811-819, 2011.
[39] A. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic Modeling Using Deep Belief Networks," *IEEE Transactions on Audio, Speech, and Language Processing,* vol. 20, pp. 14-22, 2012.
[40] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. r. Mohamed, N. Jaitly*, et al.*, "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups," *IEEE Signal Processing Magazine,* vol. 29, pp. 82-97, 2012.
[41] A. Marinescu, C. Harris, I. Dusparic, S. Clarke, and V. Cahill, "Residential electrical demand forecasting in very small scale: An evaluation of forecasting methods," in *Software Engineering Challenges for the Smart Grid (SE4SG), 2013 2nd International Workshop on*, 2013, pp. 25-32.
[42] M. Rowe, T. Yunusov, S. Haben, W. Holderbaum, and B. Potter, "The real-time optimisation of DNO owned storage devices on the LV network for peak reduction," *Energies,* vol. 7, pp. 3537-3560, 2014.

Weicong Kong (S'14) obtained his Ph.D. degree from the University of Sydney, Australia in 2017. He also received the B.E. and M.E. degrees from the South China University of Technology, Guangzhou, China, in 2008 and 2011, respectively, and the M.Sc. degree from the University of Strathclyde, Glasgow, UK, in 2009. He was the electrical engineer in Shenzhen Power Supply Company from 2011 to 2014, in charge of the development of distribution automation system, SCADA and AMI. He is now a postdoctoral research fellow with the University of New South Wales. His research interests include data analytics and deep learning in energy engineering, including non-intrusive load monitoring, load forecasting, demand response, renewable energy integration and energy market.

Zhao Yang Dong (M'99–SM'06–F'17) obtained Ph.D. from the University of Sydney, Australia in 1999. He is now with the University of NSW, Sydney, Australia. His was previously with the University of Sydney, and as Ausgrid Chair and Director of the Ausgrid Centre of Excellence for Intelligent Electricity Networks (CIEN) at the University of Newcastle, Australia. He also worked as system planning manager with Transend Networks (now TASNetworks), Australia. His research interest includes smart grid, power system planning, power system security, load modeling, renewable energy systems, electricity market, and computational intelligence. He is an editor of IEEE Transactions on Smart Grid, IEEE PES Letters and IET Renewable Power Generation. Prof Dong is a Fellow of IEEE.

Youwei Jia (S'11-M'15) received his B.Eng and Ph.D degrees from Sichuan University, China, in 2011, and The Hong Kong Polytechnic University, Hong Kong, in 2015, respectively. He is now with The Hong Kong Polytechnic University as a postdoctoral fellow. His research interests include power system security analysis, cascading failures, complex network and artificial intelligence application in power engineering.

David J. Hill (S'72-M'76-SM'91-F'93-LF'14) received the PhD degree in Electrical Engineering from the University of Newcastle, Australia, in 1976. He holds the Chair of Electrical Engineering in the Department of Electrical and Electronic Engineering at the University of Hong Kong. He is also a part-time Professor at The University of Sydney, Australia.

During 2005-2010, he was an Australian Research Council Federation Fellow at the Australian National University. Since 1994, he has held various positions at the University of Sydney, Australia, including the Chair of Electrical Engineering until 2002 and again during 2010-2013 along with an ARC Professorial Fellowship. He has also held academic and substantial visiting positions at the universities of Melbourne, California (Berkeley), Newcastle (Australia), Lund (Sweden), Munich and in Hong Kong (City and Polytechnic). His general research interests are in control systems, complex networks, power systems and stability analysis. His work is now mainly on control and planning of future energy networks and basic stability and control questions for dynamic networks.

Professor Hill is a Fellow of the Society for Industrial and Applied Mathematics, USA, the Australian Academy of Science, the Australian Academy of Technological Sciences and Engineering and the Hong Kong Academy of Engineering Sciences. He is also a Foreign Member of the Royal Swedish Academy of Engineering Sciences.

Yan Xu (S'10-M'13) received the B.E. and M.E degrees from South China University of Technology, Guangzhou, China in 2008 and 2011, respectively, and the Ph.D. degree from The University of Newcastle, Australia, in 2013. He is now the Nanyang Assistant Professor at School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He was previously with School of Electrical and Information Engineering, The University of Sydney, Australia. His research interests include power system stability, control, and optimization, microgrid, and data-analytics for smart grid applications.

Yuan Zhang (S'16) received the B.E. and M.E. degrees from Xi'an Jiaotong University, Xi'an, China, in 2010 and 2013, respectively. She is now pursuing the Ph.D. degree at the University of New South Wales, Australia. She was previously with the University of Sydney, Australia and the National University of Singapore, Singapore. Her research interests include electricity market, home energy management, statistical methods, machine learning, and their applications in power engineering.