PowerLSTM: Power Demand Forecasting Using Long Short-Term Memory Neural Network

Yao Cheng[1](✉), Chang Xu[2], Daisuke Mashima[3], Vrizlynn L.L. Thing[1], and Yongdong Wu[1]

[1] Institute for Infocomm Research, A*STAR, Singapore, Singapore
{cheng_yao,vriz,wydong}@i2r.a-star.edu.sg
[2] School of Computing Engineering, Nanyang Technological University, Singapore, Singapore
xuch0007@e.ntu.edu.sg
[3] Advanced Digital Sciences Center, Singapore, Singapore
daisuke.m@adsc.com.sg

Abstract. Power demand forecasting is a critical task to achieve efficiency and reliability in the smart grid in terms of demand response and resource allocation. This paper proposes PowerLSTM, a power demand forecasting model based on Long Short-Term Memory (LSTM) neural network. We calculate the feature significance and compact our model by capturing the features with the most important weights. Based on our preliminary study using a public dataset, compared to two recent works based on Gradient Boosting Tree (GBT) and Support Vector Regression (SVR), PowerLSTM demonstrates a decrease of 21.80% and 28.57% in forecasting error, respectively. Our study also reveals that metering/forecasting granularity at once every 30 min can bring higher accuracy than other practical granularity options.

1 Introduction

Modern smart grid is an enhanced electrical grid that takes advantage of sensing and information communication technologies to improve the efficiency, reliability and security of the power grid. Smart metering is a major improvement brought by smart grids, which facilitates real-time metering. One resulting benefit is the *power demand forecasting* based on such meter measurement, which affects the power generation scheduling and power dispatching for a future period by predicting the power demand in that period using historical data in hand.

Power demand forecasting is important for both power companies and power consumers [22]. In general, the forecasting results have different interpretations when applied to the aggregation and the individual. The aggregation forecasting, which is to predict the power demand of a number of consumption units, e.g., the apartments within an area, is more meaningful to power utility companies. Based on the aggregation demand forecasting results, they can allocate proper resources to balance the supply and demand or adjust the demand response

G. Cong et al. (Eds.): ADMA 2017, LNAI 10604, pp. 727–740, 2017.
https://doi.org/10.1007/978-3-319-69179-4_51

strategy such as dynamic pricing to shape the load so as to avoid the infrastructure capacity strain. On the other hand, individual power demand forecasting assists in the anomaly detection task in the smart metering system. Anomaly detection detects the abnormal meter measurements caused either by the unexpected meter failure or the deliberate meter manipulation by identifying those measurements that do not present a conformation to the predicted/expected values. Moreover, under dynamic pricing strategy, individual power forecasting also provides power consumers with their expected power consumption and cost in a future period, so that they can optimise their usage schedule accordingly to achieve a lower cost.

Though demand forecasting has been widely studied for years, two challenges in making accurate forecasting are still in front of us. One challenge is that even though the power demand seems like a univariate time series [28], it is subject to various influential factors which may have discriminative capability in influencing the power demand. The second challenge is that it is not trivial to chase optimal forecasting settings so as to obtain a promising results. The time granularity of metering is flexible in modern smart grids. By investigating what kinds of metering/forecasting granularities can bring an accuracy gain, it can not only provide empirical guidelines for better forecasting accuracy but also evaluate whether a model can work well with typical granularities in today's smart grid.

With the above challenges in mind, this paper proposes a power demand forecasting model named *PowerLSTM*. Firstly, we identify a set of features derived from three categories, i.e., the historical consumption data, the weather information, and the calendar information. In each category, there are a series of features that potentially and reasonably have influence on the consumers' power demand. Then, we analyse the significance of each feature and select an appropriate set of features to be used later in our forecasting model. After that, we introduce our model PowerLSTM. PowerLSTM takes advantage of the Long Short-Term Memory (LSTM) network, which is a special form of Recurrent Neural Network (RNN) with certain memory capability. In order to evaluate the effectiveness of PowerLSTM, we compare it with two representative techniques based on Gradient Boosting Tree (GBT) [5] and Support Vector Regression (SVR) [26]. For the sake of fair comparison, we implement our model and theirs as well, and evaluate all three models using a public real-world dataset. Finally, we experiment with different metering/forecasting granularities to evaluate the accuracy over different granularities that are used in practical services.

In summary, the main contributions of this paper are as below.

- We propose PowerLSTM, which, to the best of our knowledge, is the first power demand forecasting model based on LSTM that incorporates time-series features, weather features, and calendar features.
- We compare PowerLSTM with two representative models adopted in recent research works, i.e., GBT [5] and SVR [26]. In our preliminary study, PowerLSTM outperforms both models by reducing the Mean Squared Error (MSE) by 21.80% and 28.57% compared to GBT and SVR, respectively.

– We evaluate the accuracy of our model with various granularities that are typical in today's smart grid systems. The results reveal that a moderate metering/forecasting granularity at once every 30 min performs better than other granularities.

2 Related Work

Power demand forecasting has been widely studied due to its significance in power industry. The existing works can be generally classified into two categories, i.e., classic statistical models and modern machine learning algorithms.

In terms of statistical models, time-series modeling is used to capture the time-series characteristics of power demand, e.g., ARMA [14,19], ARIMA [2,7]. Hong et al. [16] adopt multiple linear regression to model hourly energy demand using seasonality (regarding year, week, and day) and temperature information. Their results indicate that complex featuring of the same information results in a more accurate forecasting. Fan and Hyndman [8] use semi-parametric additive model to explore the non-linear relationship between energy usage data and variables, i.e., calendar variables, consumption observations, and temperatures, in the short-term time period. Their model demonstrates sensitivity towards the temperature. Recently, conditional kernel density estimation is applied to power demand forecasting area [4] which performs well on dataset with strong seasonality. Time-series models are based on the assumption that the future power demand has the same or similar trend and distribution as the observed history. However, the power demand in reality is influenced by many factors in various ways. Therefore, it is essential to take these influential factors into consideration.

There are three major machine-learning algorithms used in demand forecasting tasks, namely Decision Tree (DT) [5,11,27], Support Vector Machine (SVM) [9,17,21,23,25,26], and Artificial Neural Network (ANN) [9,29]. DT is used to predict building energy demand levels [27] and analyse the electricity load level based on hourly observations of the electricity load and weather [11]. Differently, Bansal et al. [5] use an evolved version of decision tree, Boosted Decision Tree Regression (BDTR), to model and forecast energy consumption so as to create personalised electricity plans for residential consumers based on usage history. The regression based on SVM is named Support Vector Regression (SVR). There are works using SVR to forecast power consumption [25] or using it in combination with other techniques, such as fuzzy-rough feature selection [23], particle swarm optimization algorithms [21], and chaotic artificial bee colony algorithm [17]. Gajowniczek and Zabkowski choose SVM and ANN because they believe that time-series analysis is not suitable in their work since they observe high volatility in the data [9]. Yu et al. [26] uses SVM and Backward Propagation Neural Network (BPNN), whose results show that SVM offers smaller prediction errors than BPNN. Zufferey et al. [29] apply Time Delay Neural Network (TDNN) and find out that the individual consumer's consumption is harder to predict than an aggregation of multiple consumers. Very recently, Marino et al. [18] construct LSTM deep neural networks to forecast building energy load using historical consumption data. Despite the extensive research carried out in power demand forecasting area, to the best of our knowledge, there is no such work taking advantage of LSTM RNN based on features other than time series. It is promising to explore the effectiveness of such idea in power demand forecasting area, which has motivated this work.

3 Our Approach

3.1 Overview

Our approach is to forecast power demand by modeling the relationship between power demand and relevant features using the proposed PowerLSTM. Figure 1 illustrates the high-level pipeline of our approach. A recent publicly available dataset recording apartment power usage in a high frequency is used. From this dataset, we develop three categories of features that are considered relevant to the power demand. Then, we employ feature selection to remove features that are less important to reduce dimensionality and model complexity as features may not be equally effective. After that, we use the selected features to train PowerLSTM. The details of each process are explained in the subsequent subsections.

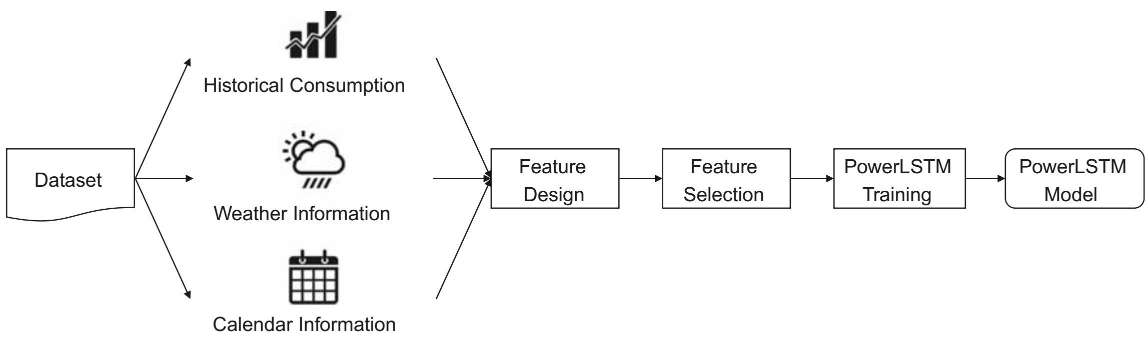


Fig. 1. Approach overview.

3.2 Power Usage Dataset

We use the publicly available power usage dataset provided by University of Massachusetts [1]. Three reasons for choosing this dataset are *(a)* when developing technology which reflects consumers' life styles and further the power consumption behaviours, a recent dataset would be beneficial to incorporate the latest power consumption characteristics; *(b)* power consumption data recorded at high frequency provides us with detailed information in finer time granularity and allows us to flexibly down-sample to explore lower granularities; and *(c)* as a public dataset, it would facilitate the further comparison with our work.

The dataset contains power usage data for 114 apartments located in Western Massachusetts for the period from year 2014 to 2016. The dataset records the power of every single apartment in fixed temporal frequency[1]. The metering frequency is once every 15 min for 2014 and 2015, and once every 1 min for 2016. Along with the power consumption data, hourly weather information, including various meteorological attributes, during the record period is available, a sample of which is shown in Table 1. In our experiment, we use the data of year 2016 because its finer granularity in recording provides us with more space for exploring influence of granularity on accuracy of power demand forecasting.

Table 1. Weather data sample.

time	**temperature**	**apparentTemperature**	**windBearing**	**dewPoint**
1451624400	36	29.75	278	24.54
summary	**humidity**	**precipIntensity**	**cloudCover**	**visibility**
Clear	0.63	0	0	10
icon	**windSpeed**	**precipProbability**	**pressure**	-
clear-night	7.94	0	1016.61	-

3.3 Feature Engineering

Feature Design. The features used by the existing forecasting models fall into three categories in terms of privacy, i.e., publicly available information (e.g., weather information), household private information (e.g., demography), and quasi-private information. The quasi-private information here is defined as privacy-related information which is known only to authorized entities. For example, the historical power consumption data acquired by a power utility company can be used to infer certain private household characteristics [3], but it is only available to the authorised personnel within the utility company.

Although household private information may have significant influence on the household power demand (e.g., more people living in the house leads to larger power demand), in this paper we limit the features to non-private information due to the following reasons. First of all, we would like to involve no household specific data in forecasting procedure other than power meter readings due to the privacy concern. Secondly, although utility companies may have access to some household private data such as locations, it is not common for them to have other private information, e.g., the number of occupants and their employment status. Thirdly, the forecasting model not based on the household specific data can be applied to larger scales easily, such as building level or area level.

We use the three categories of features in this paper, i.e., historical consumption data, weather information, and calendar information.

(a) Historical consumption data is the actual observation of the prediction target, which directly reflects the consumption pattern. Power utility companies can obtain this data by smart metering technology in smart grids.

[1] Given that the metering interval is fixed, the power is able to represent the power consumption.

(b) Weather information has influence on the power demand since some appliances (e.g., air conditions) are sensitive to weather conditions.
(c) Calendar information, such as weekday or weekend, shapes the consumers' power consumption behaviour in terms of different activities. It indicates the consumption pattern according to the calendar feature and cycle.

The features based on the above three categories are summarized in Table 2. There are $n + 18$ features in total, among which, n features are from historical consumption data, 13 are from weather information, and 5 are designed from calendar information. The historical data involves a huge number of data points which are not feasible to be fed to the model directly. Therefore, it is necessary to find out length of historical data points n that are most correlated with the target forecast value. To solve this problem, we use AutoCorrelation Function (ACF), which can quantify the correlation between time-series data points of various time lags, to find the appropriate n.

Table 2. Features for the power demand forecasting task.

Features category	Feature detail
Historical consumption data	Consumption data in past n time slots
Weather information	Summary, icon, temperature, apparent temperature, cloud cover, precip probability, precip intensity, visibility, wind speed, wind bearing, humidity, pressure, dew point
Calendar information	Day of the month, day of the week, hour of the day, period of the day (i.e., daytime and night time), is weekend (boolean value)

Feature Selection. In order to design a predictive and compact model, it is necessary to choose the features that have most significant influence on the power demand as some of the features may be redundant while some may be irrelevant. To investigate such discriminative power of features, we use feature selection to prune such redundant and irrelevant features and leave those important ones.

We use Random Forest-Recursive Feature Elimination (RF-RFE) to recursively select the optimal feature subset. It is to select a desired number of features by creating predictive models, weighing the importance of features, and eliminating those with least importance. Each recursive step is to consider a smaller set of features, and repeated till the desired number of features is reached.

3.4 Long Short-Term Memory Model

Long Short-Term Memory (LSTM) neural network [15] is a variant of RNN that is capable of learning long-term dependencies. Different from the traditional neural network that only relies on previous N histories when solving the problem, RNN allows unlimited history information to persist due to its internal loops. The network architecture of RNN makes it a prevailing choice for solving problems related to sequences. In theory, RNN should be able to learn long-term dependences, which, however, is demonstated to have practical difficulties [6]. Under this circumstance, LSTM is designed to solve the challenge of long-term dependences [13], which has demonstrated its practicality and sucess in tasks that are not solvable by other RNNs, such as continual prediction [10], speech recognition [12], language modeling [20], and translation [24]. Its capability in power demand forecasting is worth exploring due to the sequential nature of power readings.

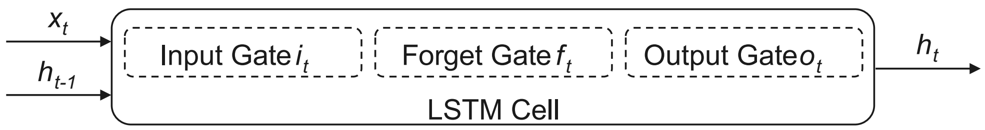


Fig. 2. LSTM cell.

The core idea of LSTM is a memory cell which can maintain the information over time controlled by various gate units. The LSTM cell as illustrated in Fig. 2 processes the information to maintain a cell status based on both current input x_t and previous output h_{t-1} (i.e., the recurrent input), then decides what information to be left and what to be passed on (i.e., h_t) by introducing gate units, i.e., "input gate", "output gate" and "forget gate". The input gate is used to control whether it allows the state in current cell to be overridden by outside information, as shown in Eq. (1),

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \tag{1}$$

where i_t is the input gate vector, σ_g the sigmoid function, x_t the input vector, W_i and U_i the parameter matrices, and b_i the bias vector. The output gate decides whether the status in the cell should affect other cells, whose formulation is shown in Eq. (2),

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \tag{2}$$

Another gate, forget gate, is introduced by Gers et al. [10] which allows the LSTM to reset its own state. It is formulated as below,

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \tag{3}$$

Finally, Eqs. (4) and (5) show how cell state c_t and output vector h_t are obtained from input gate, forget gate, and output gate,

$$c_t = f_t \odot c_{t-1} + i_t \odot \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \tag{4}$$

$$h_t = o_t \odot \sigma_h(c_t) \tag{5}$$

where $\odot$ denotes the Hadamard product, and σ_c and σ_h are the hyperbolic tangent function.

LSTM can work in a multilayer manner, each layer of which composes of multiple cells. There is a trade-off between the modeling capability and the performance efficiency. The more complicated the model is, the better capability it may have. However, this may result in an over-fitting model which performs extremely well in training set but cannot adapt to the other data of the dataset. At the same time, over-complicate model would be an inefficient model. Due to the above considerations, PowerLSTM adopts a moderate structure with two LSTM layers, so as to prevent overfitting while allowing for reasonable generalization.

4 Evaluation

This section first introduces two models in recent literature [5,26] as the baselines, and two metrics to quantify the accuracy of forecasting models. After that, we evaluate the effect of feature selection, compare PowerLSTM with two baseline models, and investigate forecasting results in different metering/forecasting granularities for evaluating the accuracy in practical use cases.

4.1 Preparation

Baseline. We choose two recent works as the baseline models in this paper. One of them adopts GBT [5] and the other one adopts SVR [26].

GBT is adopted by Bansal et al. [5] to forecast power consumption. GBT is a supervised learning predictive model which can be used for classification and regression purposes. GBT builds the model, i.e., a series of trees, in a stepwise manner. In each step, it adds one tree, while maintaining the existing trees unchanged. The added tree is the optimal tree that minimizes a predefined loss function. Basically, GBT is an ensemble of weaker prediction models, which becomes a better model.

SVM is used in the work by Yu et al. [26] to forecast power usage. SVM is a supervised machine learning algorithm for solving both classification and regression problems. SVM does classification by seeking the hyper-plane that differentiates the two classes to the largest extent, i.e., maximizing the margin. Similarly, regression using SVM, that is SVR, is to seek and optimize the generation bounds by minimizing the predefined error function. SVR supports both linear and non-linear regression. For the non-linear SVR, it transforms the data into a higher dimensional space to perform the linear separation.

Evaluation Metric. In order to evaluate the accuracy of the forecasting model, we introduce Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE). The closer the value is to zero, the more accurate the forecasting is.

MSE measures the average of the squared errors/deviations as directed by Eq. 6, where n is the total number of forecast values, A_t and F_t denote the actual and forecast value at time t, respectively.

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(A_t - F_t)^2 \tag{6}$$

Different from MSE, MAPE measures the error proportion to the absolute value. It expresses the error as a percentage and can be calculated using Eq. 7.

$$MAPE = \frac{100\%}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right| \tag{7}$$

MSE is useful in comparison experiments with identical test dataset, as it is the absolute square error value that depends on the scale of actual values. On the other hand, MAPE is more indicative in comparison between different dataset since it represents the error in a percentage manner. However, MAPE, *(a)* is not defined when A_t is zero[2]; and *(b)* has a heavier penalty on negative errors when $A_t < F_t$. Therefore, we use both MSE and MAPE to provide complementary measurements on the model accuracy.

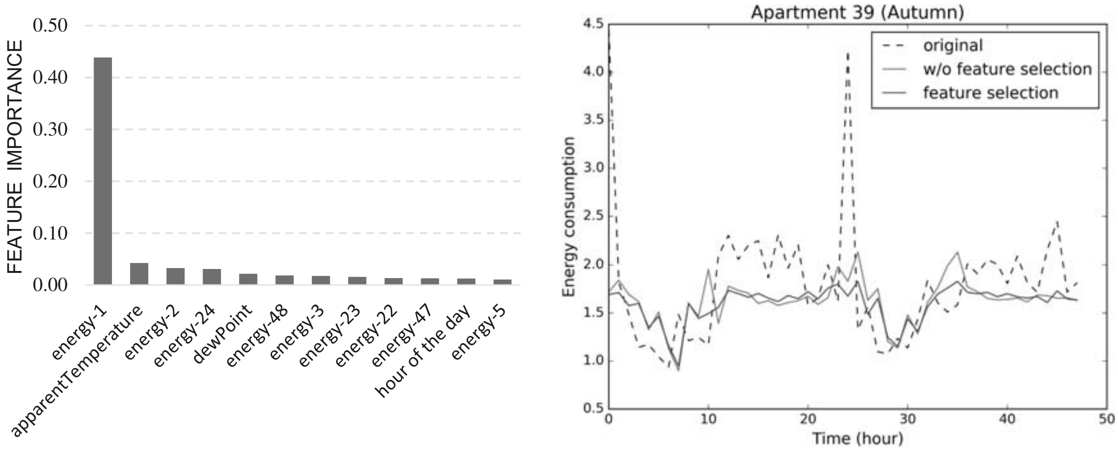
4.2 Effect of Feature Selection

As mentioned in Sect. 3.3, features may not positively contribute to the forecasting task. Redundant features may drag down the performance and irrelevant features may even disturb the prediction. This experiment investigates the contribution of each feature to the prediction, and the effect of feature selection.

We use RF-RFE to evaluate the importance weights of features in an hourly metering/forecasting granularity. Based on the ACF outcome which shows the most related number of lag is 49, we have 49 + 18 features in total. We run RF-RFE on these 67 features and obtain their importance weights. We select the top 12 important features since the importance weights after that are much less significant. Figure 3a presents the features with the top 12 importance weights. Energy-n denotes the hourly power consumption n hours prior to the target hour to be forecast. The feature with the most significant weight is the power consumption of the hour before the one to be forecast. Besides historical consumption features, three features from weather information category and calendar information category are selected into the top 12 important features list.

In order to evaluate the effect of feature selection, we separately train two models, one using all features and the other using selected features. Both models are trained on the hourly data of the first 28 days in September and tested on the following 2 days. Figure 3b shows the forecasting results of one apartment with ID 39. The results without feature selection are relatively more fluctuated than the results with feature selection. The MSE of forecasting with feature selection demonstrates an improvement by 5.44% compared to that with all features.

[2] In our experiments, we eliminate the undefined MAPE caused by a zero actual value. However, the actual power consumption values in our dataset are scarcely zero.



(a) Top 12 important features.

(b) Forecasting results with and without feature selection.

Fig. 3. Feature selection.

4.3 Comparison with Baselines

In this experiment, we compare our model with two recent works, i.e., the works from Bansal et al. [5] and Yu et al. [26] with the same training and testing data.

PowerLSTM uses a two-layered LSTM network as discussed in Sect. 3.4. The cell memory size for each layer is tuned from 160 to 200 using grid search which can exhaustively search the optimal candidate from a grid of parameter values. Similarly, the parameters for baseline models are also automatically tuned using grid search. For GBT, three parameters are tuned, i.e., number of boosting stages to perform $n\_estimators$, maximum depth of the individual regression estimators max_depth, and learning rate $learning\_rate$. Its parameter grid is constructed using $n\_estimators$: (50, 100, 150, 200, 250, 300, 350, 400, 450, 500), max_depth: (1, 2, 3, 4, 5), and $learning\_rate$: (0.001, 0.01, 0.1, 1). For SVR, three parameters, C, $kernel$, and $gamma$ are tuned. We construct the

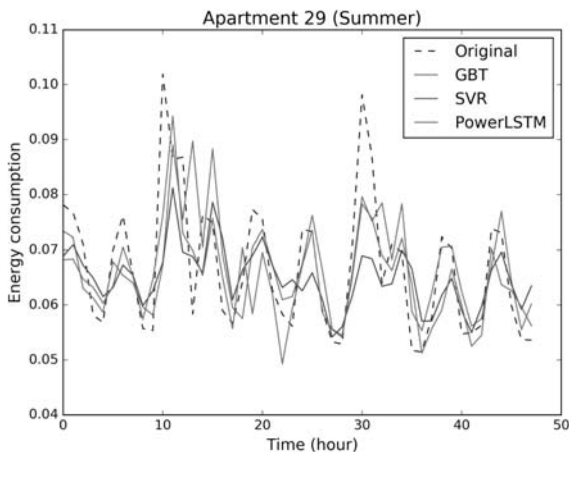


Fig. 4. Forecasting results from GBT [5], SVR [26] and PowerLSTM.

Table 3. Accuracy of GBT [5], SVR [26] and PowerLSTM.

Model	Error	
	MSE	MAPE
SVR	9.693E-05	10.391%
GBT	8.853E-05	9.512%
PowerLSTM	6.923E-05	8.935%

parameter grid using C: (0.001, 0.01, 0.1, 1), *kernel*: (rbf, linear, poly, sigmoid). *gamma* is automatically set corresponding to kernel coefficient or the reciprocal of number of features.

We use the consumption data of the first 28 days of July as the training set to train the three models, and forecast/test the expected demand for the next 2 days using the trained models. We show the forecasting results using data of apartment 29. As shown in Fig. 4, PowerLSTM is able to capture the trend as well as peaks and valleys better than both GBT [5] and SVR [26] do. Furthermore, according to Table 3, PowerLSTM brings an improvement in MSE by 21.80% and 28.58% comparing to GBT [5] and SVR [26], respectively.

4.4 Forecasting in Different Granularities

The intention of this experiment is to investigate the influence of different metering/forecasting granularities to forecasting tasks. In particular, we evaluate the forecasting accuracy under practical use cases, for instance., demand response services often require forecasting in half-hourly or hourly. In this direction, we use four different granularities when training the model, i.e., every 15 min, every 30 min, every 1 h and every 2 h. We prepare four training datasets by down-sampling the data points in the original dataset (1 min. granularity) with the four different granularities, respectively. The average value within the sample period is used in the down-sampled dataset. The model that is trained using the dataset with a lower sampling rate forecasts power demand using the same rate. To evaluate from the viewpoint of a power utility company, we show the results based on the aggregated power usage data of all 114 apartments. We use data from 1st February to 28th February as the training data and the data of the following 2 days as the testing data.

From the forecasting results shown in Fig. 5, all four models can capture the actual demand trend. Visually, the results from forecasting every 2 h are not as good as those from every 1 h and every 30 min. For quantitative understanding, we compare their MSE and MAPE in Table 4. The forecasting in 30 min. granularity demonstrates the best results in both MSE and MAPE. When the metering/forecasting granularity is low, the model may not be able to capture the consumption characteristics, as seen in the figure. On the other hand, when the granularity is high, the consumption may demonstrate more of its fluctuation and instantaneity, which may be a hindrance to the accurate forecasting task.

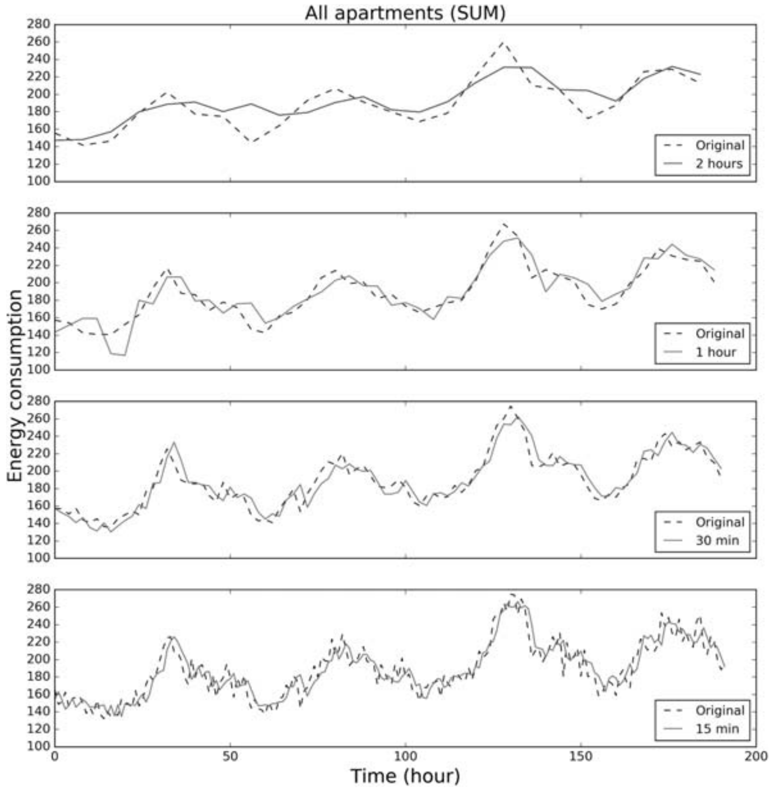


Fig. 5. Forecasting results in different granularities.

Table 4. Forecasting accuracy in different granularities.

Error	Granularity			
	15 mins	30 mins	1 h	2 h
MSE	193.218	130.982	199.767	254.400
MAPE	6.052%	4.880%	6.493%	6.773%

Having that said, PowerLSTM offers good performance when used with practically available smart meter data (e.g., in 30 min. granularity).

5 Conclusion

This paper proposes PowerLSTM, a power demand forecasting model based on LSTM, which shows an accuracy improvement comparing to two recent representative works. Further experiments in different metering/forecasting granularities reveal that the forecasting accuracy varies in different granularities and PowerLSTM can work well with typical granularities used in today's smart grid system.

Acknowledgement. This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under the Energy Programme and administrated by the Energy Market Authority (EP Award No. NRF2014EWT-EIRP002-040).

References

1. Umass smart* dataset - 2017 release. http://traces.cs.umass.edu/index.php/Smart/Smart
2. Alberg, D., Last, M.: Short-term load forecasting in smart meters with sliding window-based ARIMA algorithms. In: Asian Conference on Intelligent Information and Database Systems, pp. 299–307 (2017)
3. Anderson, B., Lin, S., Newing, A., Bahaj, A., James, P.: Electricity consumption and household characteristics: Implications for census-taking in a smart metered future. Comput. Environ. Urban Syst. **63**, 58–67 (2017)
4. Arora, S., Taylor, J.W.: Forecasting electricity smart meter data using conditional kernel density estimation. Omega **59**, 47–59 (2016)
5. Bansal, A., Rompikuntla, S.K., Gopinadhan, J., Kaur, A., Kazi, Z.A.: Energy consumption forecasting for smart meters. arXiv preprint arXiv:1512.05979 (2015)
6. Bengio, Y., Simard, P., Frasconi, P.: Learning long-term dependencies with gradient descent is difficult. IEEE Trans. Neural Netw. **5**(2), 157–166 (1994)
7. Cho, M., Hwang, J., Chen, C.: Customer short term load forecasting by using ARIMA transfer function model. In: International Conference on Energy Management and Power Delivery, vol. 1, pp. 317–322 (1995)
8. Fan, S., Hyndman, R.J.: Short-term load forecasting based on a semi-parametric additive model. IEEE Trans. Power Syst. **27**(1), 134–141 (2012)
9. Gajowniczek, K., Zabkowski, T.: Short term electricity forecasting using individual smart meter data. Procedia Comput. Sci. **35**, 589–597 (2014)
10. Gers, F.A., Schmidhuber, J., Cummins, F.: Learning to forget: continual prediction with LSTM. Neural Comput. **12**(10), 2451–2471 (2000)
11. Gładysz, B., Kuchta, D.: Application of regression trees in the analysis of electricity load. Badania Operacyjne i Decyzje **4**, 19–28 (2008)
12. Graves, A., Mohamed, A.R., Hinton, G.: Speech recognition with deep recurrent neural networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013, pp. 6645–6649 (2013)
13. Greff, K., Srivastava, R.K., Koutník, J., Steunebrink, B.R., Schmidhuber, J.: LSTM: a search space Odyssey. IEEE Trans. Neural Netw. Learn. Syst. **28**(10), 2222–2232 (2017)
14. Gross, G., Galiana, F.D.: Short-term load forecasting. Proc. IEEE **75**(12), 1558–1573 (1987)
15. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
16. Hong, T., Gui, M., Baran, M.E., Willis, H.L.: Modeling and forecasting hourly electric load by multiple linear regression with interactions. In: IEEE Power and Energy Society General Meeting, pp. 1–8 (2010)
17. Hong, W.C.: Electric load forecasting by seasonal recurrent SVR (support vector regression) with chaotic artificial bee colony algorithm. Energy **36**(9), 5568–5578 (2011)
18. Marino, D.L., Amarasinghe, K., Manic, M.: Building energy load forecasting using deep neural networks. In: 42nd Annual Conference of the IEEE Industrial Electronics Society, IECON 2016, pp. 7046–7051 (2016)
19. Mashima, D., Cárdenas, A.A.: Evaluating electricity theft detectors in smart grid networks. In: International Workshop on Recent Advances in Intrusion Detection, pp. 210–229 (2012)

20. Mikolov, T., Karafiát, M., Burget, L., Cernockỳ, J., Khudanpur, S.: Recurrent neural network based language model. In: Interspeech, vol. 2, p. 3 (2010)
21. Qiu, Z.: Electricity consumption prediction based on data mining techniques with particle swarm optimization. Int. J. Database Theor. Appl. **6**(5), 153–164 (2013)
22. Siano, P.: Demand response and smart grids-a survey. Renew. Sustain. Energ. Rev. **30**, 461–478 (2014)
23. Son, H., Kim, C.: Forecasting short-term electricity demand in residential sector based on support vector regression and fuzzy-rough feature selection with particle swarm optimization. Procedia Eng. **118**, 1162–1168 (2015)
24. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems, pp. 3104–3112 (2014)
25. Wang, J., Zhu, W., Zhang, W., Sun, D.: A trend fixed on firstly and seasonal adjustment model combined with the ε-SVR for short-term forecasting of electricity demand. Energ. Policy **37**(11), 4901–4909 (2009)
26. Yu, W., An, D., Griffith, D., Yang, Q., Xu, G.: Towards statistical modeling and machine learning based energy usage forecasting in smart grid. ACM SIGAPP Appl. Comput. Rev. **15**(1), 6–16 (2015)
27. Yu, Z., Haghighat, F., Fung, B.C., Yoshino, H.: A decision tree method for building energy demand modeling. Energy Build. **42**(10), 1637–1646 (2010)
28. Zheng, J., Xu, C., Zhang, Z., Li, X.: Electric load forecasting in smart grids using long-short-term-memory based recurrent neural network. In: 51st Annual Conference on Information Sciences and Systems, pp. 1–6 (2017)
29. Zufferey, T., Ulbig, A., Koch, S., Hug, G.: Forecasting of smart meter time series based on neural networks. In: International Workshop on Data Analytics for Renewable Energy Integration, pp. 10–21 (2016)