Week-ahead Forecasting of Household Energy Consumption Using CNN and Multivariate Data

VANITA AGRAWAL, PRADYUT K. GOSWAMI
Assam Science & Technology University Guwahati, INDIA

KANDARPA K. SARMA
Gauhati University Guwahati, INDIA

Abstract— Short-Term Load Forecasting for buildings has gained a lot of importance in recent times due to the ongoing penetration of renewable energy and the upgradation of power system networks to Smart Grids embedded with smart meters. Power System expansion is not able to keep pace with the energy consumption demands. In this scenario, accurate household energy forecasting is one of the key solutions to managing the demand side energy. Even a small percentage of improvement in forecasting error, translates to a lot of saving for both producers and consumers. In this paper, it was found out that Aggregated 1-Dimensional Convolutional Neural Networks can be effectively modeled to predict the household consumption with greater accuracy than a basic 1-Dimensional Convolutional Neural Network model or a classical Auto Regressive Integrated Moving Average model. The proposed Aggregated Convolutional Neural Network model was tested on a 4 year household energy consumption dataset and gave very promising Root Mean Square Error reduction.

Keywords—Convolutional Neural Network, Deep Learning, Household Energy consumption forecasting, Short-Term load forecasting.

Received: January 29, 2021. Revised: July 23, 2021. Accepted: August 3, 2021. Published: August 10, 2021

1. Introduction

With the advent of smart grids, intelligent homes and increasing micro-grid penetration, accuracy in Short-Term Load Forecasting (STLF) for building or industrial loads is gaining importance like never before. It becomes all the more important in the energy-insufficient countries so that the demand side response can be more efficiently managed. With the advent of deregulated electricity market scenario also in most countries, accurate prediction of load results in substantial saving for the utilities as well as the consumers. A lot of researchers are therefore scrambling to find out the best model which predicts the load with greater accuracy.

Household energy consumption forecasting is a much more difficult problem than area, bus or aggregated load forecasting due to many fluctuating variables which have to be taken into account such as performance of thermal systems, occupancy patterns etc. Their load profiles have high volatility and uncertainty. Classical approaches are no longer providing the desired accuracy and Machine Learning models are taking over [1].

From last few decades, Artificial Neural Network (ANN) models have started to be proposed for STLF. In [2] the authors explored the ANNs with back-propagation algorithm with one or two hidden layers with comparable accuracy with respect to classical methods. Guoqiang Zhang *et al.* presented a state of the art survey on ANN applications for STLF where it was summarized that ANNs were better than Regression, Auto Regressive Integrated Moving Average (ARIMA), Exponential Smoothing & other classical methods [3]. Tao Hong *et al.* [4] presented a tutorial review of probabilistic load forecasting with a comprehensive study of some representative papers in this field. In [5] the authors presented a practical neural network-based ensemble model for day-ahead building-level electricity load forecasting and showed that it outperformed Seasonal Autoregressive Integrated Moving Average (SARIMA) by almost 50%.

In [6] the authors reviewed a number of AI techniques applied for load demand forecasting for smart grids and buildings and concluded that the hybrid methods perform better.

It was shown that the over-fitting issue which is a problem when deep learning is applied, can be addressed by increasing data diversity and volume, as was shown in smart metered dataset from Ireland by batching a group of customers' load profiles into a pool of inputs [7]. Enzo Busseti *et al.* [8] explored deep learning (DL) for Time Series modeling and found out that feed-forward recurrent neural networks gave promising results.

A probability density forecasting method based on deep learning, quantile regression and kernel density estimation was proposed and the results were compared with random forest and gradient boosting machine models. The proposed deep learning approach exhibited better forecasting accuracy than others [9]. Stochastic models like Conditional Restricted Boltzmann Machine and Factored Conditional Restricted Boltzmann Machine were shown to outperform ANN, Support Vector Machine (SVM) and Recurrent Neural Networks (RNN) [10].

Deep learning was used in an unsupervised manner to extract meaningful features from raw data as model inputs to supervised learning and the results show that this methodology enhances the performance of building cooling load prediction [11].

To handle the variability and uncertainty of future load profiles, traditional Long Short-Term Memory (LSTM) based point forecasting was extended to probabilistic forecasting in the form of quantiles and was used to model both long and short-term dependencies within load profiles. Pinball loss instead of Mean Square Error (MSE) was used in training the data and forecasting results for both residential and commercial customers were tested and were found to have superior performance [12]. Mahmoud Shepero *et al.* [13] compared an especially designed log-normal process with the conventional Gaussian Processes for probabilistic load forecasting for residential customers and showed that log-normal process produced sharper forecasts. In [14], a combination of generalized extreme learning machine (GELM), wavelet processing and bootstrapping was used to forecast the electricity demand.

In the deregulated electricity market scenario, the energy imbalance, i.e. the online gap between contracted supply and actual demand need to be minimized and the market interactions based on accurate power load forecasts need to be performed. Smart meters, when fully employed, would continuously stream data and in order to take informed decisions about grid operations we need to process this data online. Petra Vrablecova *et al.* developed an online Support Vector Regression (SVR) and demonstrated its capability in accurate forecasting by comparing it with other state-of-the-art online methods [15].

Learning long-range dependencies with RNNs is difficult due to the problem of vanishing and exploding gradients. Long Short Term Memory (LSTM) was thus developed, to overcome these issues where information of the current time step could be stored and maintained to affect the LSTM output of future time steps. Weicong Kong *et al.* applied LSTM to forecast short-term residential load forecasting on a set of publicly available real smart meter data set with great improvement in results [16]. In [17] a comprehensive review of smart meter data analytics in electricity retail markets was done.

In [18] the authors reviewed the application of Deep Learning and Reinforcement Learning (RL) in Smart Grids. It was concluded that various DL algorithms e.g. Boltzmann Machine, Feed-forward Deep Networks, Convolutional Neural Networks (CNN), Recurrent Neural Networks, Long Short-Term Memory Networks & Generative Adversarial Networks (GAN) can be used for more accurate load forecasting than traditional methods.

Another aspect of building load forecasting using smart meter data, is to detect data integrity attacks where the hackers access supposedly protected data and inject false information. Jian Luo *et al.* [19] conducted an empirical study to test and benchmark the robustness of four representative forecasting models under various simulated data integrity attacks and found that Support Vector Regression (SVR) model is most robust.

Bert J. Claessens *et al.* [20] proposed a novel approach using CNN to extract hidden state-time features to mitigate the curse of partial observability. They combined CNN and a multilayer perceptron to approximate Q-values in the batch Fitted Q-Iteration using RL.

A number of approaches are hence being experimented with, so that an accurate STLF model can be found. Deep Learning methods such as RNN, LSTM, CNN [22] have gained a lot of importance in recent times [21] due its ability to solve complex problems such as computer vision, image classification etc. ANNs (nowadays also referred to as shallow networks) learn the mapping from input to output by adjusting the weights but have limited learning capability. Deep Neural Networks (DNNs) have more number of layers than ANN but they frequently encounter the problem of over-fitting and vanishing gradients which result in poor error performance, even when compared with shallow networks.

In this paper, we have hence proposed a deep learning CNN model named Aggregated Convolutional Neural Network (ACNN) model to predict the household energy consumption which results in better performance with reduction in RMSE error. To prove the efficacy of the proposed model, it has been compared with a classical method, ARIMA and a basic 1-D CNN model. Section II of the paper describes the methodology and the CNN architecture used and section III describes the experimental set-up and section IV and V present the results and conclusions respectively.

2. Proposed Work

STLF for individual buildings, households, industries etc. is a challenging problem due to the various unpredictable influencing factors such as environmental, economic and geographical. Thus, a deeper NN with multiple hidden layers and different architectures are better suited to model the STLF problem for a household than a shallow ANN. We have proposed a novel CNN architecture named ACNN to forecast week-ahead load for a single household.

2.1 Convolutional Neural Networks

ANNs receive an input and transform it through a series of hidden layers. Each hidden layer is made up of a set of neurons, where each neuron is fully connected to all the neurons in the previous layer and where neurons in a single layer function completely independently and do not share any connections. The last fully-connected layer is called the "output layer". In Convolutional Neural Networks, on the contrary, the parameters are shared i.e. weights are shared by some/all neurons in a particular feature map. There is 'local connectivity' in CNNs, where each neuron is connected only to a subset of the input, unlike ANNs where all the neurons are fully connected. This helps to reduce the number of parameters in the whole system and makes the computation faster and more efficient. This feature of CNNs helps to build deeper networks and makes them suitable for solving computer vision, image classification, object recognition and various other problems.

CNNs are comprised of various layers such as convolutional layer, pooling layer and fully connected layers. Convolution is a specialized linear operation on two functions namely input function and weighting function. The weighting function is called a "kernel" which is also a multidimensional array of weights that is updated as the algorithm learns through the iterations or epochs. The output

(s) of the convolution operation is called a feature map (s). Subsequent feature map values are calculated according to the following formula, where the input function is denoted by *f* and kernel by *h*. The indexes of rows and columns of the resultant matrix are marked with *m* and *n* respectively as shown in (1).

$$G[m,n]=(f*h)[m,n]=\sum_j \sum_k h[j,k]f[m-j,n-k] \quad (1)$$

There are mainly two different pooling mechanisms used in practice namely max-pooling and average-pooling. More precisely, the max-pooling operation, at a given position, outputs the maximum value of the input that falls within the kernel. So mathematically it can be represented as in (2).

$$h_{i,j} = \max\{x_{i+k-1,j+l-1} \ \forall\, 1 \le k \le m \ and \ 1 \le l \le m\} \quad (2)$$

Then, the elements of the feature map are run through a nonlinear activation function (ReLU). Pooling function is then used to further modify and smoothen the feature map. Pooling layers are commonly inserted in-between successive Convolutional layers. This reduces the spatial size representation by reducing the number of parameters, reduces computation and hence controls the over-fitting. Once the convolutional layers produce their outputs, the output is sent to one or more fully connected layers which is similar to the output layer of a regular NN. Learning process of the CNN is carried out using back propagation.

2.2 1-D Convolutional Neural Networks

CNNs thus have fewer weights to learn than fully connected layers and they automatically learn and generalize features from the input. They are thus able to learn highly specific features in this process.

CNNs are designed to process data that are in the form of multiple arrays [21] e.g. 1-Dimensional (D) arrays for signals and sequences, 2-D for images or audio spectrograms and 3-D for video or volumetric images. Time Series data such as energy consumption data are 1-D array.

The 1-D convolutional and pooling layers are as shown in Fig.1. $X_1, X_2, X_3, X_4, X_5, X_6, X_7$ are the inputs which after convolution with the filter of size 3, as shown in Fig.1, form the feature maps (C_1, C_2, C_3, C_4, C_5) which form the first convolutional layer. Then comes the pooling layer, which samples the feature maps and reduces the dimension (P_1, P_2). Pooling layer extracts the important features from the input.

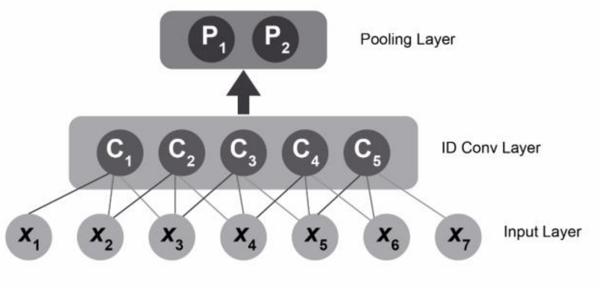

Fig. 1. 1-D CNN structure

In this work, we have used 1-D CNN for the univariate data for the basic CNN model. We have also used a separate 1-D CNN model for the multivariate data for ACNN, the proposed model. We found out that the proposed model having separate CNN sub-models for different variables outperformed the Basic CNN and ARIMA models.

3. Experimental Set-up

3.1 Dataset

The household power consumption dataset is a multivariate dataset which has the electricity consumption data from 16th December 2006 (17:24:00) till 26th November, 2010 (21:02:00) at regular intervals of 1 minute [23]. The data has 2075259 entries and 7 columns which represent global active power, global reactive power, voltage, global current intensity and 3 active energy sub-metering columns. So, the dataset provides the total active power consumption of the household as well as the active power consumptions of different utility areas of the house such as kitchen, laundry, climate controlled systems and the rest of the house. We shall be forecasting the global active power consumed in the household in this paper.

3.2 Data Processing

The data has some missing values, so before proceeding we fill in the missing values with the value at the same time, one day ago. We shall be using global active power as input and convert the time series multivariate data into a supervised learning problem. This can be done by using a 'sliding window' approach on the entire dataset. Firstly, we resample this minute-wise data into 24 hr data. Thus, we have a day-wise data now with 1442 no. of days. Secondly, we collate the data into standard weeks (which begins on Sunday and ends on Saturday) and split the weeks into training and test data. This gives us 159 standard weeks of data for training and 46 weeks of test data. These 159 weeks is a small number for the model to be trained on. So we have applied a 'sliding window' approach to augment training data as shown in Fig.2. Applying this technique, we have 1113 (159*7) weeks of training data.

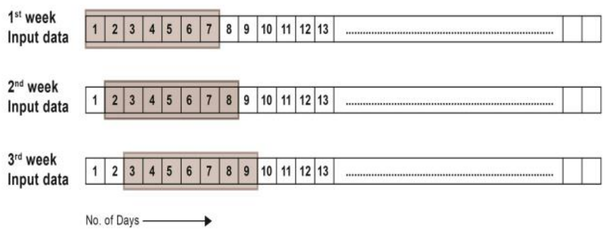

Fig. 2. Sliding Window approach

3.3 Basic CNN Model Architecture

We are using the previous week's values to forecast next week's values. So, it is a week-ahead forecast for the household energy consumption. We are using keras wrapper library with Tensorflow backend in python on Google Colab notebook.

The basic CNN 1-D model has 1 convolutional layer with 16 filters of size 3 as the first layer. It is followed by one max-pool layer with pool-size 2. Then, we have a fully connected layer with 10 neurons where activation function ReLU with 'Adam' optimizer is used. The output layer predicts the next seven days in the sequence. After a lot of trial and error, it was found out that a batch size of 4 gave good results.

3.4 Aggregated Cnn Model Architecture (Proposed)

In our proposed model, we use all the variables provided in the dataset and configure a separate CNN model for them. Later, we aggregate or concatenate all the models and forecast week-ahead total active energy consumption for the household. The aggregated CNN model has more no. of feature learning layers. The proposed ACNN model comprises of 2 feature learning layers. For each of the CNN models, we use two nos. of convolutional layers with 16 filters of size 3 and further we used ReLU activation function. A batch size of 8 gave good results with a max pool layer of pool size 2. The different sub-models are then concatenated to be fed into 2 nos. of fully connected layers which are then connected to the output layer. The output layer predicts the week-ahead values of load. The proposed ACNN architecture is shown in Fig. 3.

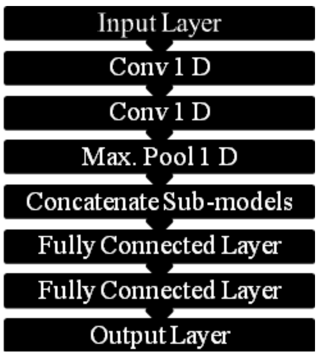

Fig. 3. Proposed ACNN architecture of one sub-model of the dataset

3.5 ARIMA Model

Looking at the Autocorrelation and Partial Autocorrelation plots of the dataset it can be concluded that there is a strong autocorrelation and that an auto regression model with 7 lag inputs can be used. So, an ARIMA model was used to get the week-ahead forecast and was compared to the basic CNN model and ACNN, proposed model.

3.6 Evaluation Metrics

We have used Root Mean Square Error (RMSE) to evaluate the forecasting accuracy between the forecasted load for the test data and the actual observed data. RMSE scores are more punishing of forecast errors than some other metrics. RMSE equation is as follows:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{n=1}^{N}(y_n - \hat{y}_n)^2}{N}} \quad (3)$$

where N is the total number of samples of the test data, y_n is the actual observed value and $\hat{y}_n$ is the predicted value.

4. Results

The following results were obtained for the Basic CNN, ACNN and ARIMA models.

4.1 Basic CNN Model

In Fig.4 line plot of the overall RMSE scores is shown for all 7 days of the week for the entire test set for Basic CNN model. The plot shows that perhaps Tuesdays and Fridays are easier to forecast than other days and probably Saturday is the hardest day to forecast.

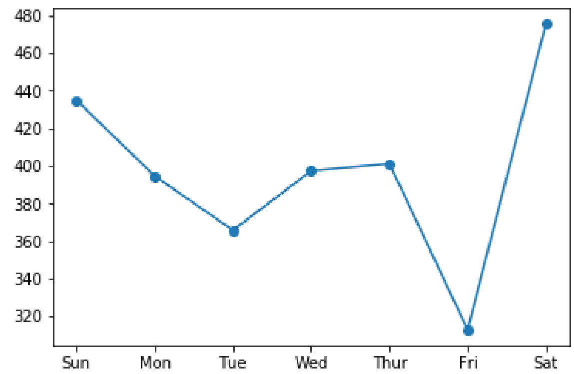

Fig. 4. Line Plot of overall RMSE scores for Basic CNN model for a week

A graph of accuracy and loss for the training data, as 300 total nos. of epochs progress is shown in the Fig.5 & Fig.6 respectively for Basic CNN model. The graphs show that training for more number of epochs enhances the accuracy and reduces the loss. A graph of real and predicted values of load for the test data is shown in Fig.7.

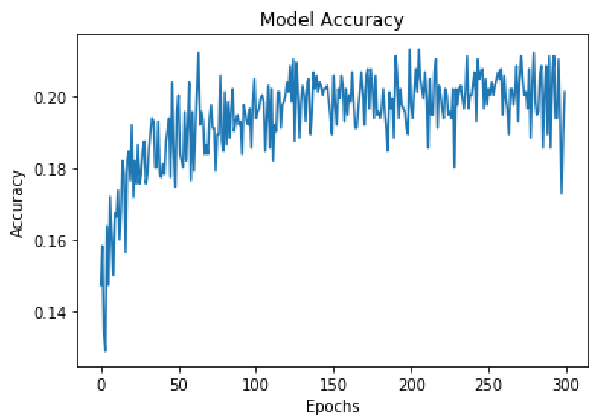

Fig.5. Accuracy Plot of Basic CNN

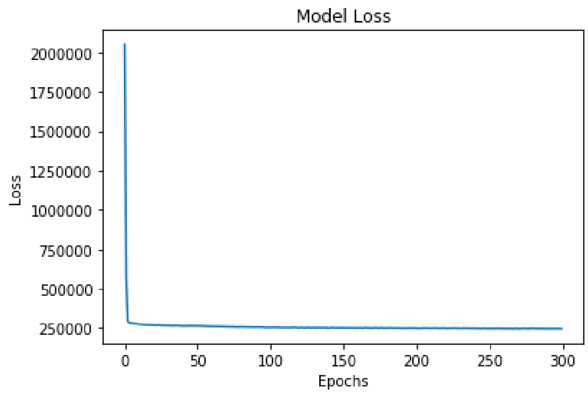

Fig.6. Loss Plot of Basic CNN

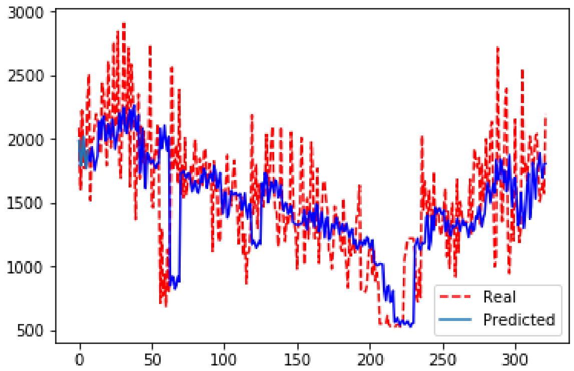

Fig. 7. Overall Real vs. Predicted Load Profile for test data for Basic CNN

4.2 Aggregated Cnn Model

In Fig.8 the line plot of overall RMSE scores for the test data is shown for all 7 days of the week for ACNN model. It can be seen that the RMSE scores improve from the basic CNN model and give better forecasting results. The use of multivariate data and deeper architecture results in better performance of the model. It creates a feature hierarchy, reduces under-fitting and increases feature learnability. An accuracy and loss plots of the training data is shown in Fig.9 & Fig.10 respectively which demonstrate that accuracy improves and loss decreases if the model is trained for longer duration but only for a certain number of epochs. In Fig.11 a graph between the real and predicted values of the load is shown for the test data.

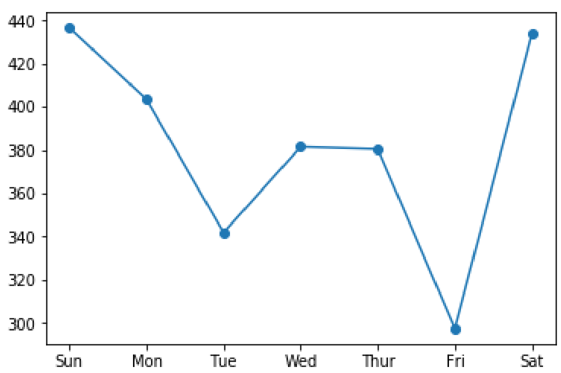

Fig. 8. Line Plot of overall RMSE scores of ACNN for a week

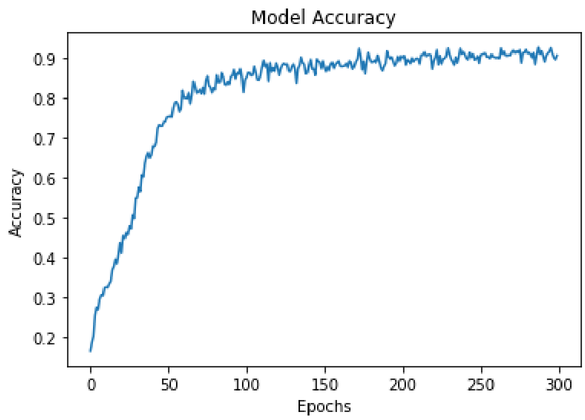

Fig.9. Accuracy Plot of ACNN

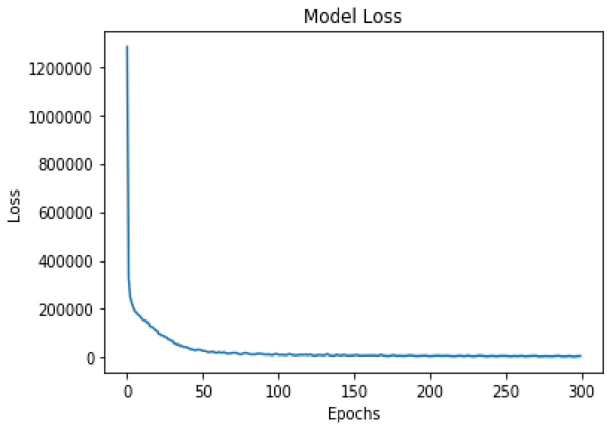

Fig. 10. Loss Plot of ACNN

The week-ahead forecast was done for all the 46 test samples and the graphs were plotted. In Fig.12 the comparison of week-ahead forecast with the real and predicted values for few sample weeks of the test data are given.

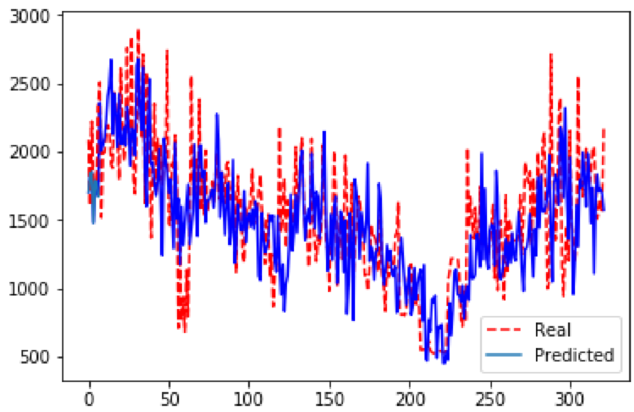

Fig. 11. Overall Real vs. Predicted Load Profile for test data for ACNN

A comparison between the graphs of the real and predicted values for both Basic CNN and proposed ACNN model shows the better RMSE scores for the proposed model.

A comparison of overall RMSE scores for basic CNN and ACNN models was done with a classical ARIMA model. It was found out that the proposed ACNN model gave the most promising results as shown in the Table I, which shows the overall RMSE scores on the testing data for various models. ACNN is thus able to follow the trend of the data showing capabilities of generalization. In Table II, a comparison of the various parameters used in Basic CNN and ACNN are shown. ACNN uses sub-models and concatenates them to get the desired results whereas Basic CNN uses univariate data. Thus the ACNN with a different architecture from Basic CNN is able to perform much better and gives very promising results for week-ahead forecast.

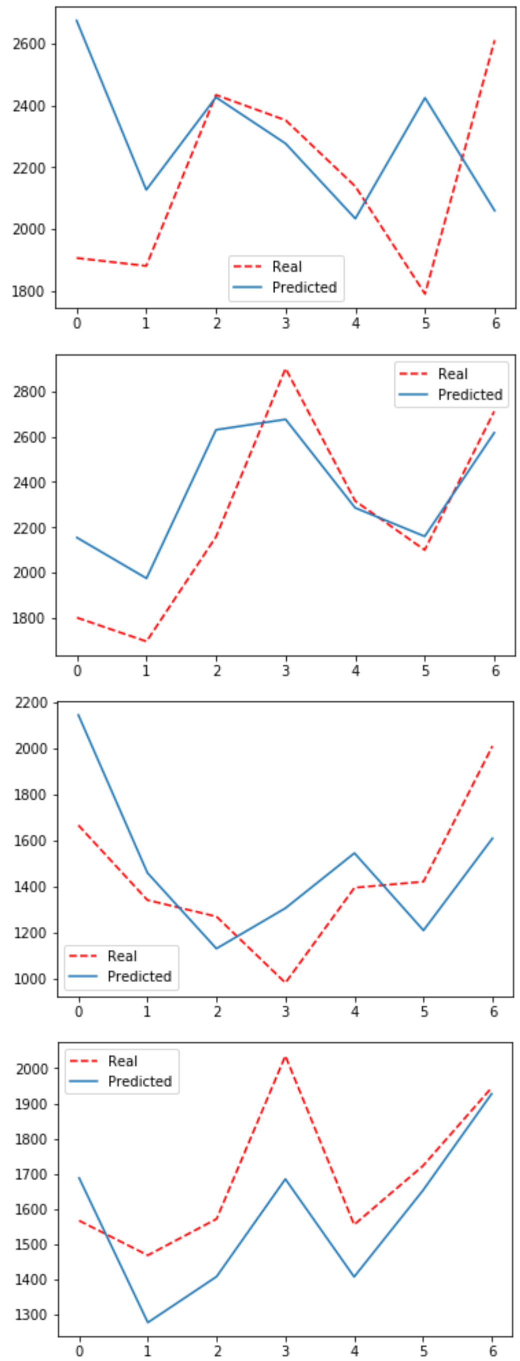

Fig. 12. Line Plots of RMSE scores for ACNN test models for few samples for a week

TABLE I
OVERALL RMSE SCORES COMPARISON OF DIFFERENT MODELS

RMSE SCORES	ARIMA	BASIC CNN	ACNN
SUN	411.7	434.8	436.5
MON	443.4	394.5	403.4
TUE	371.5	365.7	341.8
WED	405.3	397.2	381.5
THURS	417.2	401.1	380.5
FRI	309.1	312.8	297.5
SAT	462.5	475.4	434.0
OVERALL	405.655	400.145	384.939

TABLE II
PARAMETERS USED

PARAMETERS	BASIC CNN	ACNN (FOR A SUB-MODEL)
NO. OF CONVOLUTIONAL LAYERS	1	2
NO. OF FILTERS	16	16
MAX. POOL SIZE	2	2
BATCH SIZE	4	8
NO. OF NEURONS IN THE FULLY CONNECTED LAYER	10	200

4. Conclusion

This paper proposes an ACNN model to predict the household energy consumption. The model uses the multivariate data input and forms sub-models which are concatenated to forecast the week-ahead data. The dataset used, is of a single household which is split into training weekly data and testing weekly data. Previous week data is used to forecast next week's data. RMSE scores are used to measure the accuracy and it is found out that the ACNN proposed model is best suited to forecast the household energy consumption. To validate the model's effectiveness, it is compared with a basic CNN model and an ARIMA model.

Thus it is evident that, deep learning with deeper networks is a powerful tool to forecast when the data volume is quite large. In future, more efficient and different deeper networks with regularization approaches on different datasets may be developed for better forecast accuracy.

References

[1] L. Hernandez, C. Baladrón, J. M. Aguiar, B. Carro, A. J. Sanchez-Esguevillas, J. Lloret and J. Massana, "A survey on Electric Power Demand Forecasting Future Trends in Smart Grids, Microgrids and Smart Buildings," IEEE Comm. Surveys & Tutorials, vol.16, no.3, pp.1460-1495, 2014.
[2] K. Y. Lee, Y. T. Cha and J. H. Park, "Short-Term Load Forecasting Using An Artificial Neural Network," IEEE Transactions on Power Systems, vol. 7, no. 1, pp. 124-132, Feb 1992.
[3] G. Zhang, B. E. Patuwo and M. Y. Hu, "Forecasting with artificial neural networks: The state of the art," Int. Jour. of Forecasting, vol. 14, pp. 35-62, 1998.
[4] T. Hong and S. Fan, "Probabilistic electric load forecasting: A tutorial review," Int. Jour. of Forecasting, vol. 32, pp. 914-938, 2016.

[5] J. G. Jetcheva, M. Majidpour and Wei-Peng Chen, "Neural Network model ensembles for building-level electricity load forecasts," Energy and Buildings, vol. 84, pp. 214-223, 2014.

[6] M. Q. Raza and A. Khosravi, "A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings," Renewable and Sustainable Energy Reviews, vol. 50, pp. 1352-1372, 2015.

[7] H. Shi, M. Xu and Ran Li, "Deep Learning for Household Load Forecasting-A Novel Pooling Deep RNN," IEEE Transactions on Smart Grid, vol. 9, no. 5, pp. 5271-5280, Sept. 2018.

[8] E. Busseti, I. Osband and S. Wong, "Deep Learning for Time Series Modeling," CS 229 Final Project Report, Dec 14th, 2012.

[9] Z. Guo, K. Zhou, X. Zhang and S. Yang, "A deep learning model for short-term power load and probability density forecasting," Energy, vol. 160, pp. 1186-1200, 2018.

[10] 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, 2016.

[11] C. Fan, Fu Xiao and Y. Zhao, "A short-term building cooling load prediction method using deep learning algorithms," Applied Energy, vol. 195, pp. 222-233, 2017.

[12] Y. Wang, D. Gan, M. Sun, N. Zhang, Z. Lu and C. Kang, "Probabilistic individual load forecasting using pinball loss guided LSTM," Applied Energy, vol. 235, pp. 10-20, 2019.

[13] M. Shepero, D. van der Meer, J. Munkhammar and J. Widén, "Residential probabilistic load forecasting: A method using Gaussian process designed for electric load data," Applied Energy, vol. 218, pp. 159-172, 2018.

[14] M. Rafei, T. Niknam, J. Aghaei, M. Shafie-Khan and J. P. S. Catalão, "Probabilistic Load Forecasting Using an Improved Wavelet Neural Network Trained by Generalized Extreme Learning Machine," IEEE Transactions on Smart Grid, vol. 9, no. 6, pp. 6961-6971, Nov. 2018.

[15] P. Vrablecová, A. B. Ezzeddine, V. Rozinajová, S. Šárik and A. K. Sangaiah, "Smart grid load forecasting using online support vector regression," Computers and Electrical Engineering, vol. 65, pp. 102-117, 2018.

[16] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu and Y. Zhang, "Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network," IEEE Transcations on Smart Grid, vol. 10, no. 1, pp. 841-851, Jan 2019.

[17] Y. Wang, Q. Chen, T. Hong and C. Kang, "Review of Smart Meter Data Analytics: Applications, Methodologies and Challenges," IEEE Transactions on Smart Grid, vol. 10, no. 3, pp. 3125-3148 , May 2019.

[18] D. Zhang, X. Han and C. Deng, "Review on the Research and Practice of Deep Learning and Reinforcement Learning in Smart Grids," CSEE Journal of Power and Energy Systems, vol. 4, no. 3, pp. 362-370, Sept. 2018.

[19] J. Luo, T. Hong and Shu-Cherng Fang, "Benchmarking robustness of load forecasting models under data integrity attacks," Int. Journal of Forecasting, vol. 34, pp. 89-104, 2018.

[20] B. J. Claessens, P. Vrancx and F. Ruelens, "Convolutional Neural Networks For Automatic State-Time Feature Extraction in Reinforcement Learning Applied to Residential Load Control," IEEE Transactions on Smart Grid, vol. 9, no. 4, pp. 3259-3269, Nov. 2018.

[21] Y. Le Cun, Y. Bengio and G. Hinton, "Deep Learning," Review Nature, vol. 521, pp. 436-444 , May 2015.

[22] J. Wu, "Introduction to Convolutional Neural Networks," LAMDA Group, National Key Lab for Novel Software Technology, Nanjing University, 2017.

[23] https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption