

PAPER • OPEN ACCESS

Short-Term Load Forecasting in a microgrid environment: Investigating the series-specific and cross-learning forecasting methods

To cite this article: Evgenii Genov *et al* 2021 *J. Phys.: Conf. Ser.* **2042** 012035

View the [article online](#) for updates and enhancements.

You may also like

- [Optimizing Parameters of Artificial Intelligence Deep Convolutional Neural Networks \(CNN\) to improve Prediction Performance of Load Forecasting System](#)
F M Butt, L Hussain, S H M Jafri et al.

- [Load Forecasting of Central Urban Area Power Grid Based on Saturated Load Density Index](#)
Yang Huping, Tang Chengyi and Yu Meng

- [Saturated Load Forecast of Hebei Province Based on LSTM](#)
Shiyao Hu, Yang Zhao, Xiaolin Tan et al.



ECS Membership = Connection

ECS membership connects you to the electrochemical community:

- Facilitate your research and discovery through ECS meetings which convene scientists from around the world;
- Access professional support through your lifetime career;
- Open up mentorship opportunities across the stages of your career;
- Build relationships that nurture partnership, teamwork—and success!

Join ECS!

Visit electrochem.org/join



Short-Term Load Forecasting in a microgrid environment: Investigating the series-specific and cross-learning forecasting methods

Evgenii Genov¹, Stefanos Petridis², Petros Iliadis², Nikos Nikopoulos², Thierry Coosemans¹, Maarten Messagie¹, Luis Ramirez Camargo¹

¹Electric Vehicle and Energy Research Group (EVERGI), Mobility, Logistics and Automotive Technology Research Centre (MOBI), Department of Electrical Engineering and Energy Technology, Vrije Universiteit Brussel, Brussels, Belgium;

²Centre for Research and Technology Hellas - CERTH: Athens, GR

E-mail: evgenii.genov@vub.be

Abstract. A reliable and accurate load forecasting method is key to successful energy management of smart grids. Due to the non-linear relations in data generating process and data availability issues, load forecasting remains a challenging task. Here, we investigate the application of feed forward artificial neural networks, recurrent neural networks and cross-learning methods for day-ahead and three days-ahead load forecasting. The effectiveness of the proposed methods is evaluated against a statistical benchmark, using multiple accuracy metrics. The test data sets are high resolution multi-seasonal time series of electricity demand of buildings in Belgium, Canada and the UK from private measurements and open access sources. Both FFNN and RNN methods show competitive results on benchmarking datasets. Best method varies depending on the accuracy metric selected. The use of cross-learning in fitting a global RNN model has an improvement on the final accuracy.

1. Introduction

A reliable and accurate demand forecast can enhance the operation, scheduling, management and control of energy assets on a microgrid. The forecast enables matching the electricity supply with demand more efficiently with an outlook for the future consumption. In the context of a microgrid, the output of the forecast facilitates new business opportunities, such as demand response and peer-to-peer trading. On the individual building level, the non-aggregated user behaviour shows a highly dynamic and stochastic nature making forecasting a challenging task. Feed Forward neural networks (FFNN) are widely utilized in load forecasting, with ensemble and cascade models particularly finding success (see e.g. [1]). Similarly, Deep learning (DL) methods gain popularity to address this challenge due to its ability to identify relevant features in a high-dimensional feature space. Despite of being computationally intensive, DL methods, like recurrent neural networks (RNN), specifically Long Short-term memory (LSTM) typologies, are becoming more accessible and gaining more attention among forecast professionals. RNN architectures use cases exist e.g. for day-ahead forecast on real datasets both for aggregate loads [2] and individual households [3]. In this paper we compare FFNN and RNN approaches for



Content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](https://creativecommons.org/licenses/by/3.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

short-term load forecasting (STLF) within a 3-day and day-ahead horizons on the individual building level. Furthermore, this paper studies the ability of networks at cross-learning (CL). The term describes models' ability to learn from training across multiple time series in an entire data set. This allows to reduce the effect of uncertainty observed for individual users [4]. In the context of a microgrid consumption time series may be available in a limited amount, e.g. new buildings or newly-connected digital meters. In this study, we explore the benefits of CL approach. A global model is trained across a concatenated input of entire data set and evaluated on individual time series independently.

2. Methodology

2.1. Data sets

In this study electric load readings sourced from three locations are used. Two of the datasets, London and British Columbia, are accessed from open access sources. The third set, Green Energy park (GEP), is collected from private sources. A summary of key characteristics is presented in Table 1.

Dataset	London smart meter	British Columbia	Green Energy Park (GEP)
Country	UK	Canada	Belgium
No. of time series	90	28	2
Frequency	30 min	1 h	15 min
Min.Length	5737	4175	15984
Max.Length	19667	30240	44448
missing values [%]	0 - 1.72 %	0 - 1.51 %, 11.93% *	0.01 - 6 %

* Building B16 has an anomaly number of missing values

Table 1: Data Description

- London smart meter data [5]: A group of 90 buildings was selected from a bigger data set of 5,567 London households with the corresponding records of energy consumption. The original set contains information on social-economic demographic, conveyed with CACI ACORN classification [6]. For overall representation of the final selection, 5 time series were picked within each of 18 ACORN groups.
- British Columbia [7]: Hourly dataset, representing 27 residential houses in British Columbia, Canada.
- Green Energy Park (GEP): These time series correspond to multi-yearly electricity consumption records for 2 commercial buildings in a living-lab test site in Zelik, Brussels.

2.2. Pre-processing

Before using data in the training process, we apply a suitable pre-processing and feature generation. The type of pre-processing applied varies slightly between the algorithms. However, there are ideas common to both ANN and RNN approaches.

Initially, the reported missing values were filled using linear interpolation. In order to evaluate the networks on the same temporal granularity, the measurements of non-hourly frequency were re-sampled to hourly intervals with a summation (all values in kWh). Then, standard min-max normalization was utilized to scale the input and output data. An advantage of min-max scaler is that it provides uniform [0,1] scale across all features. The last step includes the train-test split. As time-series samples are not independent, the data cannot be split randomly. The data split follows the sequence logic. From the end of the series, half a year (= 4380 hours) is reserved

for test set. In time-series that hold less than a full year span of data, 20% of samples from the end of the series are reserved for test set.

2.3. Implementation

As the problem at hand is a multi-step ahead forecasting problem, a Multiple-Input Multiple-Output (MIMO) strategy is employed. An advantage of such approach is incorporating the inter-dependencies between time steps within the output window. Furthermore, MIMO strategy makes it possible to avoid potential error build-up that can occur when employing the recursive strategy, which is a sequence of one-step-ahead forecasts.

Adding calendar features boosts the network ability to learn seasonal patterns. Although some studies take an additional step in pre-processing and deseasonalize the time series data, according to [8], learning the seasonal patterns from calendar features becomes viable within the network for homogeneous series which is the case of electric load of individual buildings. The following external features were engineered:

- (i) Time of day, day of year as continuous features. The features are first converted to a fractional form in $[1, 0]$. Sin and Cos transformations were applied in order to preserve cyclical properties of the features.
- (ii) Day of week, one-hot encoded.
- (iii) Working day ticker. This is a binary feature indicating bank holidays and weekends in the countries of interest.

All models were developed using Tensorflow version 2.4.0, an open-source deep-learning platform. All necessary scripts are programmed in Python. Two lagged features are added to the input: consumption in the 72 hours and consumption in previous week for the forecasted times.

In order to limit model overfitting, we used an early stopping technique. The models are trained until the maximum number of training epochs is reached or the validation error grows. A delay in number of epochs which have the validation error growing is a patience parameter. The best epoch with smallest validation error is saved and the model is retrained on full training data with no validation split. This is done to address an issue with split into validation and train data, as highlighted in [9].

Non-model parameters set in the design of the architecture are referred as hyperparameters. The hyperparameter configuration includes the following parameters: number of layers, number of cells, learning rate, batch size and dropout rate (applicable if more than one layer). These parameters can be tuned, to achieve the best possible result. For tuning the hyperparameters of the two-stage FFNN model, an open-source Python library, keras-tuner, was used. Each of the six networks included in the model were trained and tuned separately from one another. Since the RNN training process is computationally costly, the hyperparameter optimization was run only for GEP dataset. Therefore, the final hyperparameter configuration is sub-optimal and based on a series of trials and domain knowledge. The loss function used for fitting the model on the training data optimizes the mean squared error (MSE) metric. The resulted hyperparameter configuration includes a hidden layer of 64 cells, a learning rate of $1e-3$ with 1500 batch size and no dropout rate.

2.4. Evaluation

The efficacy of the proposed algorithms needs to be assessed by comparing their results with those of an already implemented forecasting technique. For this purpose, an Auto Regressive Integrated Moving Average (ARIMA) with exogenous parameters (ARIMAX) model is created, to provide the benchmark results.

ARIMA models have been used extensively in various forecasting scenarios, not only as a benchmark, but also, as a well-established statistical method to address time series analysis problems. Adding exogenous parameters to the original ARIMA algorithm produces more accurate results, since the time series' behavior is highly dependent on external factors such as the day of the week or if it is a working day or not.

The performance is evaluated using the Root Mean Squared Error (RMSE) and the Mean Average Percentage Error (MAPE), described by the following equations:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} \frac{1}{n_O} \sum_{t=0}^{n_O-1} (\hat{y}_i[t] - y_i[t])^2} \quad (1)$$

$$\text{MAPE} = \frac{100\%}{N} \sum_{i=0}^{N-1} \frac{1}{n_O} \sum_{t=0}^{n_O-1} \left| \frac{y_t - \hat{y}_t}{y_t} \right| \quad (2)$$

where N is the number of samples in test set, n_0 is the size of the horizon, i.e. number of steps to be forecasted in each sample, and y_i , \hat{y}_i the actual and forecasted value, respectively.

3. Results

The time-series of electrical load in individual households show a high degree of uncertainty, non-linearity and anomaly days. The LSTM RNN model was trained both using individual buildings data and globally, using the concatenated input from the entire dataset.

Figure 1 shows a small sample of the time series produced by the models in contrast with the actual values for day-ahead and 3 days-ahead forecasting. It is observed that the forecasted values follow the measured ones, producing fairly accurate results. ARIMAX outputs the same results on the both day-ahead and three-days ahead horizons. Other models show a better fit at day-ahead forecasting, since there is less uncertainty and more recent lags accessible. The underestimation of peaks is attributed to regularization to avoid overfitting and the use of mean squared error as a loss function. The forecast example shows that within the observed interval RNN models output a smoother time series.

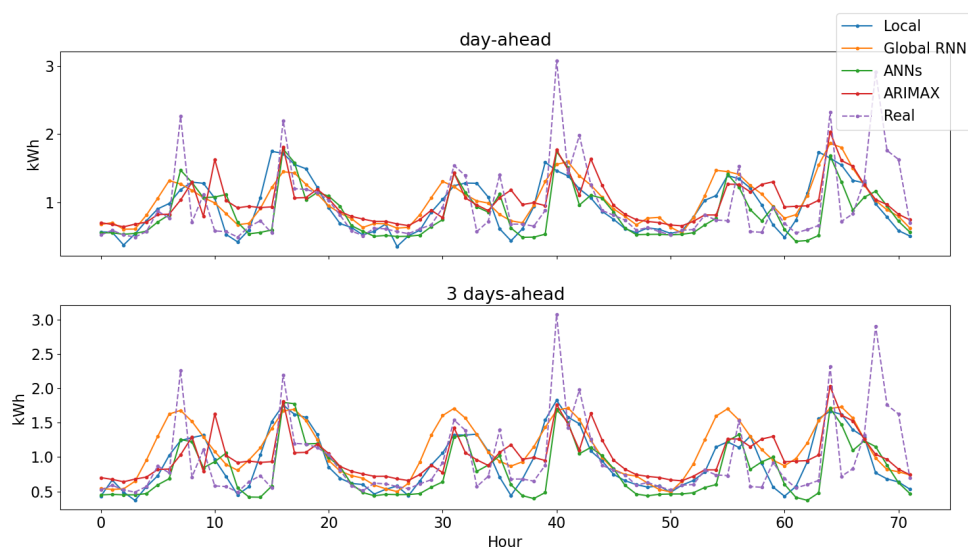


Figure 1. Time series produced by the models for day-ahead(up) and 3 days-ahead(down) forecasting

4. Conclusions

From our experiments we conclude that there is a notable difference in performance of models between day-ahead and 3 days-ahead predicting horizons. FFNN results come near to metrics of forecast produced by individual and global RNN, while outscoring them in terms of MAPE.

It was observed that CL method utilizing a global RNN model show competitive results. CL methods are advantageous for applications in microgrids, as the new assets and new measuring units can be integrated into the forecasting framework without a drop in performance. Additionally, the training of such networks requires less time than training multiple models in a series-by-series fashion.

The generalized global parameters may not successfully be applied to individual requirements of outlier time series. Therefore, hybrid and ensemble type architectures show promise and need further research for demand forecasting applications.

Acknowledgments

This research was partly funded by the H2020 RENAISSANCE project (grant number 824342) as well as by VLAIO project MAMUET (grant number HBC.2018.0529)

References

- [1] Sina Ardabili, Amir Mosavi, and Annamária R. Várkonyi-Kóczy. Advances in Machine Learning Modeling Reviewing Hybrid and Ensemble Methods. *Lecture Notes in Networks and Systems*, 101:215–227, 2020. ISSN 23673389. doi: 10.1007/978-3-030-36841-8_21. URL www.preprints.org.
- [2] Filippo Maria Bianchi, Enrico Maiorino, Michael C Kampffmeyer, Antonello Rizzi, and Robert Jenssen. An overview and comparative analysis of recurrent neural networks for short term load forecasting. *arXiv preprint arXiv:1705.04378*, 2017.
- [3] Kasun Bandara, Christoph Bergmeir, and Slawek Smyl. Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert Systems with Applications*, 140, 2020. ISSN 09574174. doi: 10.1016/j.eswa.2019.112896.
- [4] Artemios Anargyros Semenoglou, Evangelos Spiliotis, Spyros Makridakis, and Vassilios Assimakopoulos. Investigating the accuracy of cross-learning time series forecasting methods. *International Journal of Forecasting*, 2021. ISSN 01692070. doi: 10.1016/j.ijforecast.2020.11.009.
- [5] UK Power Networks. SmartMeter Energy Consumption Data in London Households, 2015.
- [6] CACI Limited. The Acorn User Guide. *The Consumer Classification*, 2014.
- [7] Stephen Makonin. Hue: The hourly usage of energy dataset for buildings in british columbia. Technical report, Simon Fraser University, 2018.
- [8] Kasun Bandara, Christoph Bergmeir, and Slawek Smyl. Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *arXiv*, 2017. ISSN 23318422.
- [9] A Suilin. Kaggle-web-traffic. <https://github.com/Arturus/kaggle-web-traffic/>, 2017. Accessed: 2021-04-30.
- [10] Rob J. Hyndman and Anne B. Koehler. Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4):679–688, 2006. ISSN 01692070. doi: 10.1016/j.ijforecast.2006.03.001.