Features and Models for Short-term Household Energy Consumption Forecast

Martin Makovec
Jožef Stefan Institute
Jamova 39, Ljubljana, Slovenia
mmakovec22@gmail.com

***Abstract*—This paper shows how to implement machine learning for load forecasting, which would lead to more effective energy generation. Researches are still trying to improve algorithms for predictions to the point where they could be released to the industry. We explore three widely applied techniques in short term load forecasting (up to 3 h). These are: Random forest regression, XGBoost regression and recurrent neural networks. A short explanation of each of these techniques, along with necessary equations, is provided. For direct comparison of these techniques UK-DALE and HUE datasets are used. It also discusses the nature of the load and the different factors influencing its behaviour.**

***Index Terms*—short-term load forecasting, long short term memory, recurrent neural network, deep learning, random forest, gadient boosted random forest, classical machine learning.**

I. Introduction

The demand for electric energy is increasing each year and with that, researchers are looking into smarter, more efficient and environmentally friendly ways of distributing electric energy. Smart grid deployments would be able to better control and balance energy supply and demand through near real time, continuous feedback about energy generation and consumption patterns. The widespread deployment of smart meters that provide frequent readings allows insight into continuous traces of usage patterns, that can be obtained through data analysis using methods such as classical machine learning and deep learning algorithms. This in turn enables better designs and triggers of demand response actions and pricing strategies, and provides input to the planning for growth and changes in the distribution network. Besides, customers may also gain better awareness of their own consumption patterns.

Load forecasting has an important role in power planning and production, thus also subject to intense research and a recent competition motivated by the effects of covid crisis on the load and load forecasting [1]. There are load forecasting models reported in literature, which can be split into two main groups, the first one being $time-series$ $(univariate)$ $models$, that use observed values from the past to form a function that models the load. The second group consists of the so-called $causal\ models$, which model the load using exogenous factors such as weather and social variables. Some of the first class models include multiplicative autoregressive models, dynamic linear or nonlinear models, threshold autoregressive models and methods based on Kalman filtering [2].

An early forecasting study on short term load forecasting (STLF) uses a multiplicative decomposition model and the seasonal autoregressive integrated moving average (ARIMA) model on Singapore's electricity data [3]. Although both time-series models can accurately predict the short-term Singapore demand, the comparison shows, that the Multiplicative decomposition model slightly outperforms the seasonal ARIMA model. Even though there are many choices regarding the use of a causal model, such as autoregressive moving average, Box and Jenkins transfer functions, structural models, optimization techniques, nonparametric regression, curve-fitting procedures and structural models, the most popular casual models are still the linear regression ones [4]. In [4] the authors use a linear regression model on electricity consumption data, provided by the local Estonian company Alexela AS, to do STLF. The model selection was motivated by specific design requirements imposed by Nord Pool and local transmission system operator data processing partner. Therefore, the focus of the paper lies on exploring the importance of the feature engineering process and discovering important factors when it comes to energy consumption profiles.

In more recent studies, artificial intelligence-based models are also explored. These models can be split into classical machine learning models [5] and deep learning models [6]. Some of the most popular classical machine learning models, when it comes to forecasting, include decision tree alghorithms and support vector machines. For instance, [5] investigated so-called kernel methods, starting from simple weighted kernel regression, all the way up to support vector machines. The prediction was carried out at different levels of aggregation (individual meters, feeder sections, distribution substations and at the system level) and done on power consumption data monitored by tens of thousands of smart meters in a medium-sized U.S. city. Additional weather data was supplied by the National Climatic Data Center and the National Oceanic and Atmospheric Administration.

The most promising deep learning appraoches to tackle STLF are recurrent neural networks (RNNs). The authors of [6] developed a long-short-term-memory (LSTM) RNN for STLF with weather features as an input. The Smart-Grid Smart-City dataset, that includes smart meter data for about 10,000 different customers in New South Wales was used. The prediction was done on residential power load, which has a greater correlation with exogenous factors such as weather or lifestyle of residents and is more irregular than regional load, which shows more seasonality.

In this paper, we report a study on developing a short term forecasting system on households from UK and Canada. We analyze the effect of various feature combinations on the forecasting accuracy for 1, 2 and 3 hours ahead. We also perform a relative performance comparison of three machine learning methods, namely Random Forests (RF), Gradient Boosted Random Forests (XGB) and Recurrent Neural Networks (RNN) and show that our results of 24% average MAPE per household are camparable with the existing state of the art where per household MAPE is between 10% and 35%.

The contributions of this paper are as follows:

- We explore a variety of different exogenous factors as well as past load data information and show their effects on model performance. Additionally we show further improvement of results upon including engineered features that capture statistical traits regarding temperature and load.
- We show that our best recurrent neural network model outperforms our best classical models, while also relying less on exogenous factors at the cost of training time.

The paper is structured as follows. Section II provides the problem statement, Section III analyzes the results provided in tables, where different features as well as model optimization is compared. Concluding remarks are drawn in Section IV.

II. Problem statement

We define our STLM regression problem as follows. Given input data consisting of time series T representing energy consumption measurments from households, weather information and other exogenous factors which affect the load, we are able to formulate a regressor Φ, which is a function that can predict energy consumption (E) of households in the future.

$$\Phi(T) = E$$

The regressor Φ is realized using classical machine learning as well as deep learning tehniques and the UK-DALE [7] and HUE [8] datasets.

A. Dataset summary

The UK-DALE (Domestic Appliance-Level Electricity) contains appliance-by-appliance power demand of 5 UK homes recorded approximately once every 6s. House 1 has almost 5 years of recordings while others have around half a year. To develop 1-3 hour ahead forecasting we downsampled the data to every hour by averaging (when predicting 1, 2, 3 hours ahead). To complement the UK-DALE dataset we also used MIDAS Open: UK daily temperature dataset, which contains maximum and minimum temperature for each half-day period[1].

The HUE dataset contains aggregated power consumption data of 28 houses in British Columbia, Canada. Energy consumption is recorded with an hourly frequency, with most houses having three years of consumption history. Weather data from the nearest weather station and metadata regarding houses is also included. Some of the missing temperature samples were linearly interpolated to perserve as much data as possible.

B. Feature Engineering

Starting from the available data, we noticed that in addition to energy and weather measurements, also meta-data such as type of household and orientation was available, therefore we engineered several feature sets to study their influence on the quality of the forecasting.

Due to space constraints, we report the results for a selection of three feature sets as follows:

Set1 HUE - Consists of hourly samples of current energy consumption of the house. It is used as a baseline for comparison to other feature sets.

Set1 UK-DALE - Also consists of hourly energy consumption samples.

Set2 HUE - Consists of *energy, temperature, part_of_day, part_of_year, weekend, facing, EV, RU, type.*

Set2 UK-DALE - UK-DALE data lacks temperature measurements for every sample, as well as meta-data such as the presence of electrical vehicle (EV), geographical orientation (facing) and residential unit information (RU), therefore this set consists only of : *energy, part_of_day, part_of_year, weekend, type.*

Set3 HUE - Additionally we also engineered some statistical features regarding energy and temperature, therefore this set consists of *energy, energy_day_mean, energy_day_max, energy_day_min, house_energy_mean, energy_diff, energy_day_diff, temperature, temperature_day_mean, temperature_day_max, temperature_day_min, temperature_diff, part_of_day, part_of_year, weekend, facing, weekend, EV, RU, type.*

Set3 UK-DALE - Similarly, we engineered features on UK-DALE consisting of: *energy, energy_day_mean, energy_day_max, energy_day_min, house_energy_mean, energy_diff, temperature_day_max, temperature_day_min, energy_day_mean, part_of_day, part_of_year, weekend, type.*

C. Selected Techniques

We consider the following set of techniques: Random Forest Regressor (RF), Recurrent neural network (RNN), XGBoost Regressor (XGB). RF combines many different decision trees. The final result is the average of all results given by individual decision trees. XGB works very similarly but it also implements different boosting techniques to more optimally split the nodes. The RNN algorithm is widely used in forecasts, because it can take into account many data samples from the past to form a more solid prediction, that is more immune to random disturbances. The main problem of recurrent neural networks is their short memory, which means they are influenced the most by the last sample of the sequence they process. We use a so called long-short term memory recurrent neural network (LSTM RNN), which battles this problem. We present

[1] https://data.ceda.ac.uk/badc/ukmo-midas-open/data/uk-daily-temperature-obs/dataset-version-202107/

results for the following set-ups selected after hyperparameter optimization. For the classical models the optimization was done with the help of GridSearchCV while the deep learning model was optimised through trial and error. For the deep learning model we used the *ReLu* activation function with the *Adam* optimizer.

RF default - Random forest model with default configuration, i.e. 100 estimators, with minimum number samples required to split an internal node of 2 and minimum number samples required to be at a leaf node of 1

RF optimized - With random forest we also kept the number of estimators at 100. We increased minimum samples required to split the node to 20 and mininimum number of samples required at a leaf node to 40.

XGB default - XGBoost model with default configuration, i.e. 100 estimators, minimum child weight of 1 and maximum depth of 3

XGB optimized - We kept the number of estimators at 100 and increased the minimum child weight to 50 and maximum depth to 21

RNN default - The recurrent neural network at first consisted out of a LSTM layer with 16 neurons, followed by a dense layer with 32 neurons and a dense output layer with a single neuron. We ran the model for 10 epochs, while considering only the previous sample to get a baseline result.

RNN optimized - Final model consisted out of only an LSTM layer with 32 neurons and an output dense layer with one neuron. This structure worked well, without any overfit compared to more complex models with multiple layers and an emphasis on previous sample information - having only an LSTM layer. We increased the number of epochs to 20, that is when the loss stopped improving. We also increased the number of previous samples considered to 5 which has shown to be optimal. It gave the model a good boost in performance, while not being too computationally heavy.

D. Evaluation Metrics

To ensure credible results k-fold cross-validation was used and the source code is publicly available[2]. For evaluating the performance of the predictor we use mean absolute error (MAE) and mean absolute percentage error (MAPE):

$$MAE(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} |y - \hat{y}_i| \tag{1}$$

$$MAPE(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} |\frac{y - \hat{y}_i}{y}| \tag{2}$$

where $\hat{y}$ is the predicted and y the real value.

III. Results

In this section we will first analyze the effect of feature selection, followed by the effect of model selection on the performance of STLF 1-3 hours ahead. With increasing time horizons predictions are generally worse.

[2]https://github.com/MMakovec/IJS_models

A. Effect of feature selection

It can be seen from Tables I and II that by enriching the training features the average performance across both datasets grows from 46,17% for Set1 to 36,19% for Set3. Although features from UK-DALE data brought a significant improvement in performance, we are still missing some key features for hourly predictions, like temperature data for every hour. This explains the poor performance of 46,52% average MAPE of classical models when predicting on Set3 of Uk-DALE data. We can see the RNN model performs with an average MAPE over all prediction horizons of 28,13% on the same data and 37,23% even without any additional features, so the lack of features does not affect it as much. That is because past energy consumption already has a big correlation with future energy consumption and the RNN model is able to take into account multiple samples from the past to form a better prediction than classical models, which cannot do that.

B. Effect of model selection

Turning our attention to Tables III and IV. We can see that average performance gain from model optimization is 6,46% MAPE, which is not as drastic as 9,98% MAPE with addition of features, but still noticable. Although classical models perform badly on UK-DALE because of missing features, the RNN model showed even better performance on UK-DALE than it did on HUE data, with an average 28,13% MAPE of the opzimited model on UK-DALE compared to average 31,06% MAPE on HUE. The reason for that would be less variety within the UK-DALE dataset, as it only includes five different houses compared to twenty eight different houses of HUE. Each individual house in the UK-DALE dataset also has a larger number of samples, leading to better training of the RNN model. Overall RF and XGB models performed very similarly with the best performing model being the RNN. It can perform good predictions without needing a lot of extra features as long as the sample size is large enough.

IV. Conclusion

To summarize, this paper has presented the performance of state of the art machine learning models in the area of forecasting system load with prediction times of 1 to 3 hours. It shows the difference between classical machine learning, deep learning and their seperate use cases. Classical machine learning offers a quicker way to train with minimal data available, while deep learning combined with a large dataset offers better performance. Although RNN needs a lot more samples to properly train, the paper showed it to be less feature dependent than RF and XGB. Besides that, it compared standard features used in load forecasting and their correlation to the energy consumption. Energy consumption feature proved to be the most important for the RNN model, while other features like temperature and time features became a lot more important when predicting with RF and XGB models.

TABLE I
EFFECT OF FEATURE SELECTION ON STLF ON THE UK-DALE DATASET.

Feature set	Metric	RF			XGB			RNN		
		1h	2h	3h	1h	2h	3h	1h	2h	3h
Set1	MAE [Wh]	289,34	350,70	376,40	295,15	355,07	380,46	252,71	310,02	328,30
	MAPE [%]	45,016	60,286	67,385	45,915	60,588	65,588	30,955	38,665	42,087
Set2	MAE [Wh]	280,70	331,99	360,45	297,00	346,82	374,15	244,34	265,20	273,93
	MAPE [%]	43,142	55,614	63,867	43,588	55,199	63,200	30,029	34,159	34,896
Set3	MAE [Wh]	**270,46**	**313,46**	**338,57**	**272,89**	**312,85**	**336,03**	**233,42**	**257,67**	**268,12**
	MAPE [%]	**36,481**	**49,349**	**55,409**	**36,512**	**47,446**	**53,973**	**24,023**	**28,685**	**31,672**

TABLE II
EFFECT OF FEATURE SELECTION ON STLF ON THE HUE DATASET.

Feature set	Metric	RF			XGB			RNN		
		1h	2h	3h	1h	2h	3h	1h	2h	3h
Set1	MAE [Wh]	317,41	460,59	491,94	315,57	458,70	490,07	282,18	358,31	394,85
	MAPE [%]	36,881	41,568	44,275	36,123	42,159	44,313	34,322	44,383	50,640
Set2	MAE [Wh]	319,07	432,90	454,76	294,20	400,04	419,83	271,39	324,88	348,28
	MAPE [%]	35,807	40,968	41,162	35,927	41,524	43,013	34,595	34,816	46,157
Set3	MAE [Wh]	**275,37**	**334,83**	**365,44**	**275,24**	**349,76**	**358,14**	**248,07**	**284,79**	**298,07**
	MAPE [%]	**30,108**	**33,722**	**33,517**	**30,128**	**33,730**	**33,544**	**29,780**	**30,709**	**32,682**

TABLE III
EFFECT OF MODEL SELECTION ON STLF ON THE UK-DALE DATASET.

Model	Metric	RF			XGB			RNN		
		1h	2h	3h	1h	2h	3h	1h	2h	3h
Deafault	MAE [Wh]	295,65	337,15	363,27	274,41	319,87	345,25	244,51	261,93	268,14
	MAPE [%]	44,285	55,434	62,571	42,574	55,004	62,522	30,197	39,360	41,681
Optimized	MAE [Wh]	**270,46**	**313,46**	**338,57**	**272,89**	**312,85**	**336,03**	**233,42**	**257,67**	**268,12**
	MAPE [%]	**36,481**	**49,349**	**55,409**	**36,512**	**47,446**	**53,973**	**24,023**	**28,685**	**31,672**

TABLE IV
EFFECT OF MODEL SELECTION ON STLF ON THE HUE DATASET.

Model	Metric	RF			XGB			RNN		
		1h	2h	3h	1h	2h	3h	1h	2h	3h
Deafault	MAE [Wh]	300,31	378,41	383,88	290,65	396,35	410,95	264,99	324,73	350,34
	MAPE [%]	36,283	40,251	40,404	35,119	39,416	39,785	30,694	35,525	36,647
Optimized	MAE [Wh]	**275,37**	**334,83**	**365,44**	**275,24**	**349,76**	**358,14**	**248,07**	**284,79**	**298,07**
	MAPE [%]	**30,108**	**33,722**	**33,517**	**30,128**	**33,730**	**33,544**	**29,780**	**30,709**	**32,682**

ACKNOWLEDGMENT

This work was funded by the Slovenian Research Agency under the Grant P2-0016. I would like to thank my mentors Carolina Fortuna[3], Gregor Cerar[4] and Blaž Bertalanič[5] for their support with this work.

REFERENCES

[1] F. Mostafa, B. Jethro, W. Yi, M. Stephen, S. Wencong, and Z. Hamidreza, "Day-ahead electricity demand forecasting compeition: Post-covid paradigm," *IEEE Open Access Journal of Power and Energy*, vol. 9, pp. 185–191, 2022.

[2] C. Deb, F. Zhang, J. Yang, S. E. Lee, and K. W. Shah, "A review on time series forecasting techniques for building energy consumption," *Renewable and Sustainable Energy Reviews*, vol. 74, pp. 902–924, 2017.

[3] J. Deng and P. Jirutitijaroen, "Short-term load forecasting using time series analysis: A case study for singapore," in *2010 IEEE Conference on Cybernetics and Intelligent Systems*, 2010, pp. 231–236.

[4] M. Spichakova, J. Belikov, K. Nõu, and E. Petlenkov, "Feature engineering for short-term forecast of energy consumption," in *2019 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*. IEEE, 2019, pp. 1–5.

[5] P. Mirowski, S. Chen, T. K. Ho, and C.-N. Yu, "Demand forecasting in smart grids," *Bell Labs technical journal*, vol. 18, no. 4, pp. 135–158, 2014.

[6] 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 Transactions on Smart Grid*, vol. 10, no. 1, pp. 841–851, 2017.

[7] K. Jack and K. William, "The uk-dale dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes," *Scientific Data*, vol. 2, 2015. [Online]. Available: https://www.nature.com/articles/sdata20157

[8] S. Makonin, "HUE: The Hourly Usage of Energy Dataset for Buildings in British Columbia," 2018. [Online]. Available: https://doi.org/10.7910/DVN/N3HGRN

[3] https://e6.ijs.si/people/carolina-fortuna
[4] https://e6.ijs.si/people/gregor-cerar
[5] https://e6.ijs.si/people/blaz-bertalanic