Energy Prediction under Changed Demand Conditions: 
Robust Machine Learning Models and Input Feature Combinations

Thomas Schranz¹, Johannes Exenberger¹, Christian Mølødrup Legaard², Ján Drgoňa³, 
Gerald Schweiger¹

¹Graz University of Technology, Graz, Austria
²Aarhus University, Aarhus, Denmark
³Pacific Northwest National Laboratory, Richland, WA, USA

Abstract
Deciding on a suitable algorithm for energy demand prediction in a building is non-trivial and depends on the availability of data. In this paper we compare four machine learning models, commonly found in the literature, in terms of their generalization performance and in terms of how using different sets of input features affects accuracy. This is tested on a data set where consumption patterns differ significantly between training and evaluation because of the Covid-19 pandemic. We provide a hands-on guide and supply a Python framework for building operators to adapt and use in their applications.

Key Innovations
With this paper, we contribute to the state of the art in building energy forecasting by assessing the performance and the robustness of four machine learning algorithms (linear regression, random forest, fully-connected neural network, and recurrent neural network) with various sets of input features. We analyze models in terms of their ability to predict short-term energy demand in a building where the consumption patterns differ significantly between training and test data because of the Covid-19 pandemic. We provide guidelines for practitioners by

- examining how different lookback and prediction horizons influence the accuracy and robustness of the machine learning models for single-step energy demand prediction
- benchmarking models using additional input features, such as weather data, against models predicting future energy demand from past consumption values only.
- examining the potential of integrating water consumption data for data-driven energy prediction.

In order not to bias the comparison, the models were not hyper-parameter-tuned to our use case. Instead, the Python machine learning framework, as well as the data used in the experiments described here is published on Github (Link will be supplied after review). This allows researchers and practitioners to reproduce the results presented in this paper, adapt the framework for their purposes and/or develop and improve models to match their requirements (e.g. in terms of accuracy).

Practical Implications

- Use features engineered from date and time (time of day, weekday, holiday), as it efficiently increases model performance and robustness.
- Random forest provides a simple solution for prediction tasks with adequate accuracy also in scenarios of changed demand patterns.
- Integrating water consumption is not generally recommended to increase robustness, as it is only beneficial in specific forecasting scenarios.

Introduction
The European Union proposed a comprehensive set of measures to drastically cut greenhouse gas emissions by 2030 and become the first climate-neutral continent by 2050 (European Commission and Climate Action DG, 2019). With a share of 78%, the energy sector is the largest contributor of emissions in the European Union (EU) and Iceland (European Environmental Agency, 2020). Driven by socioeconomic changes, such as the trend towards larger homes and the broad availability of energy-consuming entertainment, the buildings sector has become the largest contributor to the global energy demand (Allouhi et al., 2015), accounting for 40% of the energy consumption in the EU (European Commission, 2016).

It is apparent that building designers and operators play a principal role in the reduction of energy-related greenhouse gas emissions. Besides traditional energy saving measures, such as improving thermal insulation or retrofitting heating systems, hot-water boilers and lighting systems, Chwieduk (2003) proposed the introduction of environmentally-friendly energy technologies in the form of automation and data analysis to control, optimize and reduce energy demand as early as 2003. Other strategies in-
clude increasing the share of volatile renewable energy sources (Mathiesen et al., 2015), (Chwieduk, 2003), reducing energy demand by encouraging active user participation (Schweiger et al., 2020), (Schranz et al., 2020), and the use of energy services such as demand response (Meyabadi and Dehimi, 2017) or model predictive control (Mariano-Hernández et al., 2021). Many of these strategies rely on accurate prediction of future energy demand, which remains a key research interest.

Advances in embedded computing and cloud technologies provide researchers with large amounts of operational data such as detailed historical energy consumption. Among others, Rätz et al. (2019) have highlighted the capabilities of data-driven modeling techniques and their applications to building control and building energy optimization.

In this paper, we compare the performance of four different data-driven models (linear regression, a random forest regression ensemble, a fully-connected neural network, and a recurrent neural network) that predict future energy demand from past observations in a scenario where measures against the Covid-19 pandemic drastically changed consumption patterns.

We benchmark models predicting from past energy consumption values only against models with additional input features, such as water consumption data or weather information in terms of performance and generalization ability. Besides, we examine if the models are able to provide reasonable predictions for the time during the pandemic even when they were trained on consumption data from before the pandemic.

We aim to provide general observations and guidelines for practitioners to help them develop data models for energy forecasting, rather than present another set of models that provide good results for the data set used in the paper but might not work well for their particular use case or data set. We test models that provide straightforward implementation without the necessity for complex hyper-parameter tuning. We use what we consider "default" hyper-parameter settings, that are typically found in the literature, and/or default choices in the Python libraries we use. Consequently, we focus on identifying the most relevant input features for prediction performance and robustness, and how different models are affected by the choice of feature sets, lookback and prediction horizon.

Building Energy Forecasting: Related Work

There exists a large body of literature in the domain of data-driven load forecasting for building energy prediction. Latest reviews of state of the art approaches can be found in Wei et al. (2018), Fathi et al. (2020), and Sun et al. (2020). Machine learning models, such as decision trees, support vector machines (SVM) (Jain et al., 2014), nonlinear regression (Wei et al., 2019), and deep neural networks (Somu et al., 2021), all of which have been applied successfully in the domain of energy forecasting for more than a decade now (Zhao and Magoulès, 2012), are among the most popular approaches.

The majority of models found in the literature use meteorological data, features engineered from date and time, occupancy and historical energy consumption as inputs. Models described in recent studies predict the energy demand for a myriad of building types, such as educational buildings, offices, commercial and industrial complexes as well as residential buildings on various levels of aggregation, ranging from single buildings over city blocks to whole districts. Key objectives are the prediction of total energy demand, electric power consumption or heating/cooling demand. Even when focusing on use cases that are comparable in methodology and objective to the present paper, i.e. the application of data-driven methods to predict electric energy demand of a single office building (academic/mixed-use), a substantial amount of contributions can be identified.

Wang et al. (2018) apply a random forest ensemble to predict hourly energy usage of two university buildings and validate the performance against a regression tree and a SVM model. The model input consists of weather data, including temperature, wind speed and solar radiation, information about the time of day, the weekday, etc. and estimated occupancy data. Hourly occupancy is approximated using class schedules and the number of students registered for each class, the number of staff members working in the building and time tables. The authors find that occupant behavior is a key contributor to uncertainty and that the significance of input variables, likely because of changing operational conditions, varies for different semesters.

Walker et al. (2020) compare the performance of seven machine learning algorithms for time series analysis of energy consumption, including random forests, SVMs and artificial neural networks. The authors use the models to predict hourly energy demand on individual building level and on a building cluster consisting of 47 commercial buildings. They base their choice of features on their understanding of building operation and choose weather information, categorical date and time features (day of week, time of day) and autoregressive past consumption values (past day and past week) for day ahead predictions and acknowledge that the feature selection has a non-trivial influence on model accuracy.

Gaussian process regression, a machine learning prediction approach with relatively low computational complexity is presented in Zeng et al. (2020). The authors use standardized regression coefficients as well as what they call domain knowledge of energy computation to chose weather conditions and information from occupancy schedules to predict energy
consumption in six large commercial buildings. The potential of transfer learning in artificial neural networks for 24h ahead building energy demand prediction is investigated in Fan et al. (2020), using data of 507 non-residential buildings including offices, schools and university facilities. The authors conclude that transfer learning can be a useful approach when insufficient training data is available.

Method

In this section we describe the use case and its boundary conditions and approximate mathematical formulation of the prediction process. We characterize the data, its sources and the preprocessing steps we applied to it. We explain the data models, including their inputs and outputs, outline the training process for the neural networks and list the metrics we used.

Use Case Description

In this paper we investigate approaches to predict hourly electric energy usage of a five-floor, mixed-use academic building at the Graz University of Technology (TUG) accommodating offices, seminar rooms, laboratories and a lecture hall.

The data models we developed predict a single energy consumption value from a finite set of previously observed values, i.e. the forecasting problem is formulated as a generic regression problem (Bontempi et al., 2013). There are two parameters to this process: i) the lookback horizon, i.e. the number of past observations the prediction is based on and ii) the prediction horizon, i.e. how far the model predicts into the future.

In Figure 1 the blue graph represents the time series development, with the solid line representing past and the dashed line representing future values (with respect to the current time $t_0$). In this example, four values from the past in conjunction with the current value (marked with green circles) constitute the basis for projecting the value (marked as a red triangle) of time series two steps into the future. This corresponds to a lookback horizon of four and a prediction horizon of two.

In the baseline models, predictions are inferred from a single type of previously observed values. In the dynamical systems approach, this is considered as estimating the future system state from previous and current system states - i.e. predictions of the hourly energy consumption are based on previous consumption values only. Each input vector $x$ is of size $[n, 1]$, where $n = \text{lookback horizon} \ l + \text{one current state}$. The output $y$ is a single scalar value, i.e. the expected energy consumption $c^t$ at time step $t + p$, where $p$ is the prediction horizon.

$$x = \begin{bmatrix} c^{t-2} \\ \vdots \\ c^t \end{bmatrix}, \quad y = c^{t+p}$$

To improve prediction accuracy it is possible to use additional information that is expected to be correlated with the energy consumption to describe the system state at any given point in time. Examples are information about the outside temperature, whether it was day or night, the hourly water consumption in the building, or the number of registration for a lecture hall or seminar room located within the building. Using multiple characteristics to describe the system state turns the scalar components in the input vector into vectors. Consequently, the input vector turns into an input matrix $X$ of size $[n, m]$, where $m$ corresponds to the number of characteristics, subsequently denoted features $f_i$. The model output $y$ remains unchanged, as we are still only interested in the future energy consumption.

$$X = \begin{bmatrix} f_0^{t-l} & \ldots & f_m^{t-l} \\ \vdots & \ddots & \vdots \\ f_0^t & \ldots & f_m^t \end{bmatrix}, \quad y = c^{t+p}$$

Recurrent neural networks are designed to handle two dimensional inputs with a time axis and a feature axis. For the other models, i.e. linear regression, random forest and fully-connected network, this matrix has to be flattened to a vector $x$, effectively removing any distinction between different features of the same time step and same features of different time steps.

$$x = \begin{bmatrix} f_0^{t-l} & \ldots & f_m^{t-l} \\ f_0^t & \ldots & f_m^t \end{bmatrix}, \quad y = c^{t+p}$$

Be aware that the energy consumption values may or may not be part of the state description. In the subsequent section we outline the data available to us.

Data

In this section we describe the nature and sources of the data we used in the experiments. The Buildings and Technical Support (BATS) department at the TUG manages buildings and infrastructure on three campuses. In the course of their operations they capture real time data from various types of smart sensors (Schranz et al., 2020). For the experiments described here, the BATS department provided us with...
consumption sequences from smart water and smart energy meters located in a mixed-use academic/office building.

**Energy and water consumption data** is available from May 5, 2019 to July 21, 2020 in one-hour intervals. It is worth noting the special character of this timeframe as the months from March 2020 to July 2020 coincide with the Covid-19 pandemic. Due to the measures imposed by the Austrian federal government and the university directorate access to the academic facilities was restricted, classroom teaching suspended and staff members were directed to work from home whenever possible. This provides the opportunity to investigate how successful the models are in predicting the energy demand given the significant change in consumption behavior because of the restrictions. This is of particular interest because the boundary conditions stay the same, i.e. training and test data are still from the same building but occupancy and occupant behavior are different. All models were trained using data recorded prior to the measures coming into effect and tested on data from periods where the restrictions were in place.

**Occupancy data** is approximated through schedules exported from the TUG resource management system. The dataset contains the dates and registrations for all events and courses that take place in the lecture hall located in the building.

**Weather data** is obtained through a web API (http://at-wetter.tk) that provides access to the open data collection published by the Austrian "Zentralanstalt für Meteorologie und Geodynamik" (ZAMG). The data was captured at the Graz Airport, which is located about 8 km from the campus site and contains, among other metrics, hourly temperature measurements and the time of sunrise and sunset.

The **date and time features** are engineered from the timestamps the energy consumption data is indexed with. Plots show that energy consumption amplitude correlates with the weekday, the occurrence of public holidays and that consumption is, on average, notably higher during the semester. For each time step we calculate the weekday, a Boolean feature encoding whether it coincides with a public holiday, a Boolean feature whether it is during the semester and the time of day. The time of day and the numeric representation of the weekday assume periodic values, 0 to 23 (because consumption is sampled hourly) and 0 to 6, respectively. To capture periodicity both features are encoded with a sine/cosine pair (Drezga and Rahman, 1998).

**Models**

We developed four data models, two statistical models (linear regression and a decision tree ensemble) and two neural networks (a fully-connected sequential model and a recurrent network). The entire framework (for data preparation, preprocessing, training, benchmarking and plotting) and all models were implemented using Python 3. For the statistical models we used the machine learning library sklearn (https://scikit-learn.org/stable) and for the neural networks we used the TensorFlow (https://www.tensorflow.org) implementation of the Keras API (https://keras.io).

The **linear regression (LR)** model is relatively simple and requires no parameter settings. As the input samples have to be provided as one-dimensional vector, multi-feature input matrices are flattened along the time axis in a pre-processing step.

The **random forest (RF)** is built from 100 estimators, with no restrictions on maximum depth. A minimum number of two samples is required for splitting internal nodes, at each leaf node there has to be at least one sample. The mean squared error (MSE) is used as splitting criterion and all input features are used in the split. The random forest requires input samples to be vectors, i.e. multi-feature inputs have to be flattened along the time dimension.

The **neural networks** are built as a stack of layers with inputs of variable size. They accept either a vector (only one scalar value per lookback step) or a \([n, m]\) matrix of multiple features for multiple time steps (although the fully-connected network flattens it internally) and produce variable size outputs. With this architecture, it is possible to either predict multiple system characteristics at one time step or one characteristic at multiple time steps in the future. However, to obtain comparable results to the ones generated by the statistical methods, we used exactly one output, i.e. the hourly energy consumption at one single point in the future defined by the prediction horizon.

All networks are trained on batches of 72 samples using an RMSprop optimizer with a learning rate of 0.001, \(\rho\) and \(\epsilon\) parameters of 0.9 and \(1e^{-7}\) respectively. We set the maximum number of training epochs to 200 but implemented early stopping to avoid overfitting. The training data is split such that the last 20\% of the samples (subsequently called validation data) in the set are used to monitor convergence. Training is stopped when the mean squared error on the validation data does not decrease for 30 consecutive training epochs.

The **fully-connected neural network (NN)** flattens the input along the time dimension, i.e. it uses the same input as the two statistical methods. The flattened input passes through two layers consisting of 64 rectified linear units (ReLU) and is output in a single dense layer with one unit.

The **recurrent neural network (RNN)** uses an architecture designed specifically for time series prediction. The network sequentially applies transformations along the input’s time axis, remembering infor-
Table 1: Errors of energy demand prediction for all models using different feature combinations. The bold numbers indicate the best performing models for every feature combination, while the gray cells indicate the overall best performing combination of model and features.

<table>
<thead>
<tr>
<th>Features</th>
<th>Baseline Model: Energy consumption</th>
<th>Energy consumption, weather</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Prediction horizon</td>
<td>1h</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CV-RMSE MAPE R²</td>
<td></td>
</tr>
<tr>
<td>LR</td>
<td>0.14 0.24 0.28 0.30 0.31</td>
<td>0.14 0.23 0.27 0.29 0.32</td>
</tr>
<tr>
<td></td>
<td>6.5 12.1 16.7 20.7 23.0</td>
<td>6.8 13.0 17.8 21.0 22.7</td>
</tr>
<tr>
<td></td>
<td>0.78 0.40 0.16 0.07 -0.03</td>
<td>0.78 0.44 0.22 0.10 -0.04</td>
</tr>
<tr>
<td>RF</td>
<td>0.14 0.23 0.25 0.27 0.28</td>
<td>0.14 0.22 0.24 0.25 0.27</td>
</tr>
<tr>
<td></td>
<td>5.8 9.5 12.8 16.5 19.0</td>
<td>5.5 8.7 10.7 14.2 15.5</td>
</tr>
<tr>
<td></td>
<td>0.78 0.44 0.33 0.24 0.16</td>
<td>0.79 0.47 0.39 0.32 0.22</td>
</tr>
<tr>
<td>NN</td>
<td>0.15 0.22 0.26 0.27 0.29</td>
<td>0.14 0.21 0.24 0.29 0.29</td>
</tr>
<tr>
<td></td>
<td>6.4 10.4 12.5 15.3 16.2</td>
<td>6.4 9.7 12.5 20.3 18.3</td>
</tr>
<tr>
<td></td>
<td>0.77 0.51 0.30 0.25 0.13</td>
<td>0.79 0.55 0.41 0.11 0.15</td>
</tr>
<tr>
<td>RNN</td>
<td>0.15 0.24 0.28 0.29 0.29</td>
<td>0.14 0.22 0.25 0.30 0.29</td>
</tr>
<tr>
<td></td>
<td>5.9 10.7 17.3 21.6 18.0</td>
<td>6.1 11.6 12.8 18.3 20.3</td>
</tr>
<tr>
<td></td>
<td>0.77 0.39 0.18 0.13 0.20</td>
<td>0.79 0.48 0.34 0.08 0.11</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Features</th>
<th>Energy consumption, datetime</th>
<th>Energy consumption, water</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Prediction horizon</td>
<td>1h</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CV-RMSE MAPE R²</td>
<td></td>
</tr>
<tr>
<td>LR</td>
<td>0.14 0.22 0.25 0.26 0.27</td>
<td>0.14 0.23 0.28 0.30 0.32</td>
</tr>
<tr>
<td></td>
<td>6.7 12.5 16.0 16.8 16.4</td>
<td>6.6 12.4 17.7 22.2 25.1</td>
</tr>
<tr>
<td></td>
<td>0.79 0.47 0.32 0.28 0.27</td>
<td>0.79 0.43 0.20 0.08 -0.08</td>
</tr>
<tr>
<td>RF</td>
<td>0.14 0.19 0.21 0.20 0.23</td>
<td>0.15 0.25 0.25 0.24 0.29</td>
</tr>
<tr>
<td></td>
<td>5.5 7.7 9.0 9.9 11.9</td>
<td>5.9 9.3 11.8 15.2 19.4</td>
</tr>
<tr>
<td></td>
<td>0.79 0.61 0.55 0.57 0.44</td>
<td>0.78 0.45 0.34 0.37 0.12</td>
</tr>
<tr>
<td>NN</td>
<td>0.14 0.22 0.25 0.27 0.28</td>
<td>0.14 0.21 0.25 0.27 0.29</td>
</tr>
<tr>
<td></td>
<td>6.9 14.3 16.7 20.3 16.6</td>
<td>6.0 9.1 12.6 15.5 16.0</td>
</tr>
<tr>
<td></td>
<td>0.80 0.49 0.35 0.23 0.20</td>
<td>0.79 0.55 0.35 0.26 0.14</td>
</tr>
<tr>
<td>RNN</td>
<td>0.14 0.20 0.23 0.29 0.22</td>
<td>0.14 0.24 0.27 0.27 0.30</td>
</tr>
<tr>
<td></td>
<td>6.2 8.9 13.2 19.9 12.5</td>
<td>6.2 11.4 16.2 18.2 23.8</td>
</tr>
<tr>
<td></td>
<td>0.79 0.56 0.43 0.11 0.51</td>
<td>0.78 0.38 0.21 0.22 0.06</td>
</tr>
</tbody>
</table>

Results

All models were trained separately using six different input feature combinations and lookback horizons of 12, 48 and 72 hours to predict hourly energy demand one, three, six, 12 and 24 hours into the future. The following feature combination were tested:

- energy demand
- energy demand + weather conditions
- energy demand + date/time features
- energy demand + water consumption
- energy demand + occupancy
- combination of all features above

Results show that models using either occupancy or a combination of all features perform poorly. It appears that occupancy data, approximated by the number of registrations for courses and events held in the building does not properly reflect the actual number of occupants. Especially, because the registration management system does not seem to be reflecting the changes caused by the Covid-19 restrictions. Consequently, these two combinations are omitted in the subsequent analysis. Table 1 contains the results for all other feature combinations for all models with a lookback horizon of 24 hours.
Figure 2 shows the prediction accuracy on the test data for models with a lookback horizon of 24 hours in the first row and 72 hours in the second row. It can be seen that for one-hour-ahead prediction all models perform reasonably well for any choice of input features and lookback horizon. Conversely, 24-hour ahead prediction seems to be difficult for all models. For the three- to 12-hour-ahead prediction, differences in performance between the models and the feature choices are most noticeable. Changing the lookback horizon from 24 to 72 hours does not generally improve accuracy, although the RF seems to benefit slightly from a longer lookback when it uses energy consumption and the date/time features as input. Including water consumption causes no significant change in accuracy for neither model or choice of lookback.

Discussion

Experiments show that the choice of input features and lookback horizon has a varied influence on the different models. We subsequently discuss the findings for each model in detail. There are however, some general observations from all experiments:

- Lookback should be at least 24h.
- All models/features show similar performance for one-hour-ahead prediction.
- Using weather data or water consumption data does not increase accuracy in general (except for fully-connected NN).

Linear regression:
- Using date/time features significantly increases prediction performance compared to using previously observed energy consumption only.
- Changing the lookback from 24 to 72 hours causes no noticeable improvement.

Random forest:
- Using date/time features significantly increases prediction performance compared to using previously observed energy consumption only.
- A longer lookback horizon improves performance.
- Results indicate that RF has the best generalization ability of all models we tested and that it adapts well to the changes in the energy demand patterns.
- RF is the model that is least sensitive to the choice of input features.

Fully-connected neural network:
- With a longer lookback horizon, using hourly water consumption slightly increases prediction performance compared to using previously observed energy consumption only.
- When using previous consumption values alone the NN shows similar performance to the RF model.
- The choice of feature combination and lookback horizon interact with each other.
- The NN does not adapt well to changed demand patterns.
- The NN is very sensitive to the choice of input features.

Recurrent neural network:
- Date/time features significantly increase performance if the lookback horizon and the prediction
horizon are 24 hours.

- Performance does not increase with longer lookback horizons.
- The RNN does not adapt well to changed demand patterns.
- The RNN is very sensitive to the choice of input features.

Conclusion

With the development in machine learning algorithms for time series analysis, practitioners are provided with a myriad of choices. Deciding on a suitable algorithm for energy demand prediction in a building is non-trivial and depends on the availability of data. In this paper we compared four machine learning models, commonly found in the literature, in terms of their generalization performance and in terms of how using different sets of input features affects accuracy.

We evaluated the models on three metrics, the coefficient of variance of the root mean square error (CV-RMSE), the mean absolute percentage error (MAPE) and the coefficient of determination ($R^2$), all of which are widely used to assess building energy demand prediction performance in the literature.

Besides previous consumption values, we used features engineered from date and time (time of day, weekday, holiday), weather data (outside temperature and daylight hours), estimates for occupancy and water consumption data as model inputs. We trained all models on data captured between May 2019 and March 2020, before the onset of the Covid-19 pandemic in Austria, and tested them on data recorded between March and July 2020, where strict measures imposed by the federal government gravely affected energy consumption patterns. The energy and water consumption data was recorded in an academic office building in hourly intervals. We benchmarked different lookback and prediction horizons for the models, ranging from 12 to 72 hours for the lookback and 1 to 24 hours for prediction.

Results show that using features engineered from date and time affects prediction performance most significantly, regardless of the choice of model and lookback horizon. Additionally, we found that simple models, such as linear regression and random forests perform very well both in terms of generalization ability and robustness with respect to the choice of input features and lookback horizon. Especially the random forests showed exceptional generalization performance for all choices of input features. Conversely, the neural networks performed well when predicting from previous consumption values alone, but were sensible to the choice of inputs. It stands to reason that this issue could be addressed with regularization techniques, such as dropout layers, L1 or L2 regularization. Besides, we found that the choice of input features and the choice of lookback horizons for the neural networks interacted with each other. Consequently, we did not find neural networks models to adequately fulfill the requirement of working well without extensive testing and tweaking. Using water consumption data to predict energy demand seemed to improve performance of the neural networks, however, we did not find the results to be conclusive. The benefit of integrating water consumption data into energy prediction models has to be investigated in more detail in follow-up research.

References


