Journal of Physics: Conference Series

PAPER • OPEN ACCESS

Application of machine learning to predict the thermal power plant process condition

To cite this article: M M Sultanov *et al* 2022 *J. Phys.: Conf. Ser.* **2150** 012029

View the article online for updates and enhancements.

Application of machine learning to predict the thermal power plant process condition

M M Sultanov, I A Boldyrev and K V Evseev

Volzhsky Branch of Moscow Power Engineering Institute, ul. Lenina 69, Volzhsky, Russia

kirillevseyev@gmail.com

Abstract. This paper deals with the development of an algorithm for predicting thermal power plant process variables. The input data are described, and the data cleaning algorithm is presented along with the Python frameworks used. The employed machine learning model is discussed, and the results are presented.

1. Introduction

Building a digital system for predicting process conditions leading to thermal power equipment malfunctions is crucial for providing high operation efficiency and safety of thermal power plants. Malfunctions can be both direct technological (vibrations, turbine metal temperature, etc.) and general (equipment condition, fuel-consumption rate etc.). To predict such parameters, it is proposed to use time series forecasting techniques since the data in this case are an array of technological values together with timestamps [1–3].

The goal of this study is to develop an online algorithm for forecasting thermal power plant process variables to improve the plant efficiency and reliability.

The input data were retrieved from a historical database and contain process variables of a power plant turbine set, such as the sharp steam flow, the temperature and pressure behind the boiler, and the temperature of the various metal parts of the turbine set. The raw data contain approximately 41000 entries over 25 days.

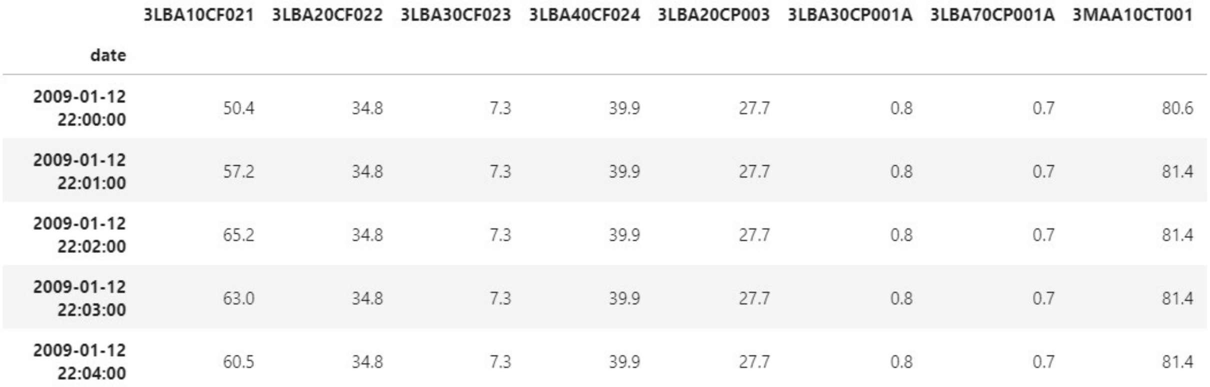
The algorithm was developed using Python, which is a high-level programming language that provides a flexible environment for data analysis and machine learning. The following major Python frameworks and libraries were used to process the data:

- Numpy (for working with array data);
- Pandas (for working with panel and time-series data);
- Scikit-learn (for preprocessing data to fit a machine learning model);
- Matplotlib (for data visualization and presentation);
- Keras (for building and fitting machine learning models).

2. Data preparation

The raw input data are saved as a comma-separated values (CSV) file, where each column represents a parameter value and the index is in date/time format. The column names correspond to the parameter name signal markings, which, though less conveniently representable, are easier to access when addressing specific values in the data table. Signals are represented using a dictionary which contains markings as keys and parameter names as values, making it possible to get a user-friendly description of a particular signal marking when analyzing and visualizing data.

The first step in analyzing process variables is to join all parameters by timestamps using outer merge. Since the data is stored in the software-and-hardware historical database at different time intervals, i.e., some parameters can be written more often, the next step is to fill the empty fields with the last known states, assuming that the parameter value has not been updated yet [4]. However, after the join, the time interval between the entries may be different. Therefore, to use the timeseries in forecasting, it is necessary to time sample the data at 1-min intervals, so that the timestamp delta for each field is the same and equals one minute. An example of a formatted data table is presented in Figure 1.



	3LBA10CF021	3LBA20CF022	3LBA30CF023	3LBA40CF024	3LBA20CP003	3LBA30CP001A	3LBA70CP001A	3MAA10CT001
date								
2009-01-12 22:00:00	50.4	34.8	7.3	39.9	27.7	0.8	0.7	80.6
2009-01-12 22:01:00	57.2	34.8	7.3	39.9	27.7	0.8	0.7	81.4
2009-01-12 22:02:00	65.2	34.8	7.3	39.9	27.7	0.8	0.7	81.4
2009-01-12 22:03:00	63.0	34.8	7.3	39.9	27.7	0.8	0.7	81.4
2009-01-12 22:04:00	60.5	34.8	7.3	39.9	27.7	0.8	0.7	81.4

Figure 1. Process variable data table

Each parameter can be represented as a plot where the X-axis is the timestamp and the Y-axis is the parameter values. The turbine set process variable corresponding to the metal temperature in the steam inlet zone in the lower half of the high-power cylinder (HPC) is shown in Figure 2.

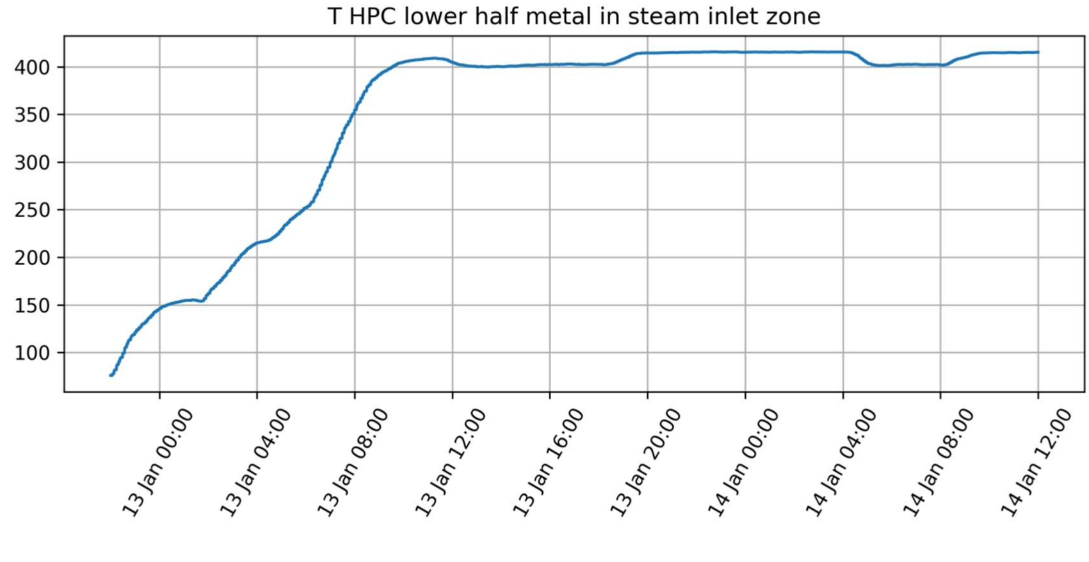


Figure 2. Process variable plot

The parameter presented above is one of the target parameters to be monitored during thermal power operation, especially when starting or stopping a turbine set. The plot represents a transient process, a case where forecasting process variables is important for providing reliable and efficient operation.

3. Process variable forecasting

Process variable forecasting is needed to prevent malfunctions and improve the power equipment efficiency. To make a forecast, historical data close to the period of prediction are required. Any forecasting algorithm should be validated before it can be used to predict future values, so that when forecasting a process variable, it is necessary to split the input data into training and validation sets. In fitting a model, the training set is used to find the optimal model weights and the validation set is used to estimate the error in predicting new values. The validation set error is calculated at the end of an epoch end and is a metric that indicates whether the model is overfitted and allows making a decision when to stop fitting the model. The study validation set is 10% of the dataset used to predict future values.

To predict the future values are predicted using a forecast window which contains a number of previous values. The size of the forecast window depends on the parameter. The prediction interval corresponds to the number of future values to be forecasted using the values within the forecast window. To predict the HPC metal temperature within 15 to 30 min, it is suggested to take into account the values for the last 30 min. to 1 hour. The forecast dataset should include several hours before the forecast point.

To forecast process variables, it is proposed to build a machine learning model consisting of two long short-term memory (LSTM) blocks (Figure 3). This structure is due to the fact that values of a process variable depend on its previous states. Moreover, it is proposed to consider not only the regressor, the process variable itself, but also include other variables, predictors that are shifted in time. Therefore, to predict future values, it is necessary to use the values of both the regressor and predictors within the forecast window. The proposed model takes into account the values for the last 60 min. and predicts the future values of the regressor for 30 min, i.e., its output contains 30 values.

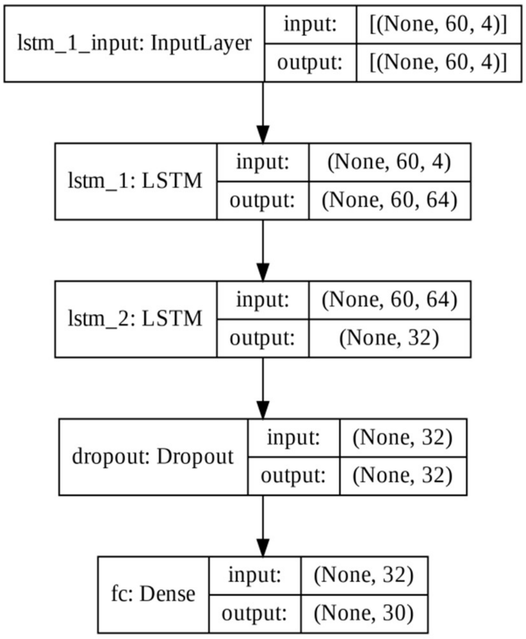


Figure 3. Forecast model structure

Before using data to fit the model, they should be normalized to avoid large and small errors and to speed up the training phase. To fit the model, both the regressor and predictors should be split into arrays in which each element is a window containing the corresponding values. The error between the ground true and predicted values is calculated using the mean absolute deviation (MAD). During model training, it is necessary to avoid over fitting. For this, at the end of each epoch, the validation loss is compared with the minimum loss obtained and the best model weights are stored in memory. After the model fitting, the best weights are restored to make predictions.

It should be noted that the model should be fitted each time before forecasting using the last data obtained. This is due to the fact that the data should be close to the forecast point to make correct predictions and new data has their own mean and standard deviations, so that reverse normalization may return a false result. To provide the algorithm performance the model should be fitted at least every 5 minutes. The model predictions are used to evaluate the process and take actions to improve the unit performance. The loss metric graph during the training phase is presented in figure 4, where each epoch corresponds mean squared error for both training and validation data sets.

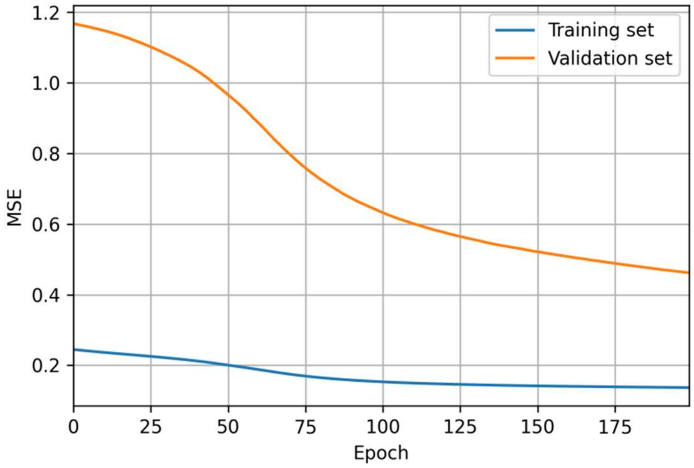


Figure 4. Training loss graph

The results for the metal temperature in the steam inlet zone in the HPC lower half are presented in Figure 4. The dashed line corresponds to the ground truth, and the solid line to predictions. The vertical line at 6:00 separates the training data on the left from the prediction interval on the right. To plot the predictions for training dataset the first values of the array of predictions that returns the model were taken. The reason to plot these predictions is model performance visualization over both training and testing datasets.

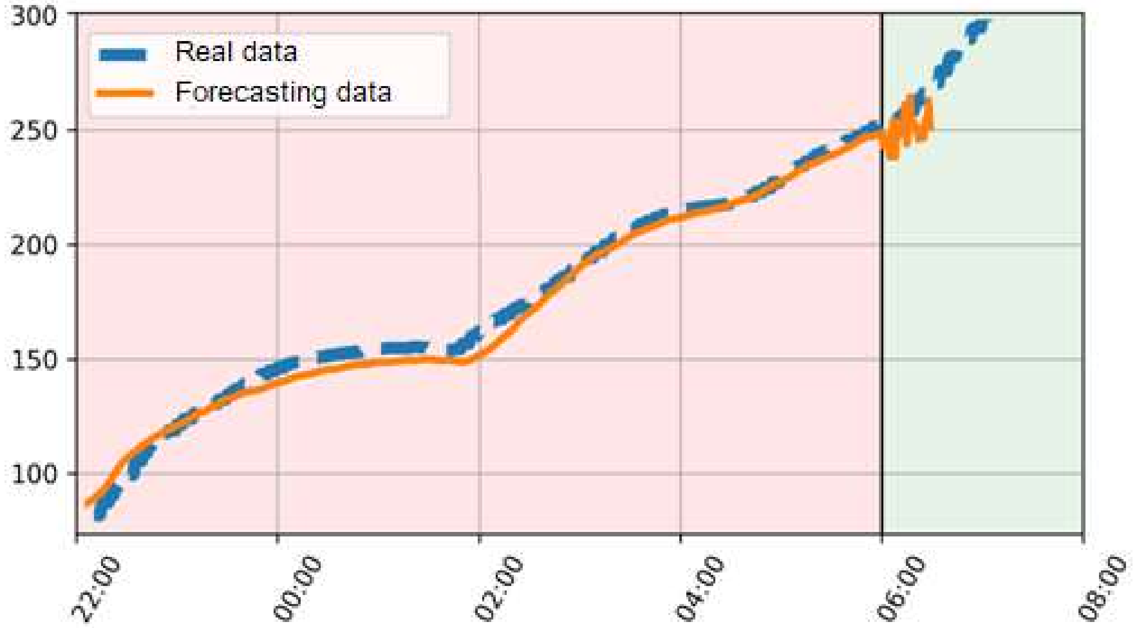


Figure 5. Process variable forecast

4. Conclusions

The proposed algorithm allows forecasting process variables taking into account the previous state of both the regressor itself and predictors; therefore, the prediction is based on multivariate time series forecasting. The forecasting pipeline provides automation of the future value prediction, which allows one to analyze data online, retrieve the last acquired values, and return predictions within a certain time interval as feedback. The use of the proposed solution allows improving the reliability and efficiency of thermal power plants by preventing inefficient generating unit operation.

Acknowledgements

The research is funded by Russian Federation public contract FSWF -2020-0025 «Technique development and method analysis for ensuring power system object security and competitiveness based on the digital technologies».

References

[1] John Harlim, Shixiao W. Jiang, Senwei Liang, Haizhao Yang. Machine learning for prediction with missing dynamics, Journal of Computational Physics, 2021, Vol. 428, 109922

[2] Wang, Z. Zhao, B. Guo, H. Tang, L. Peng, Y. Deep Ensemble Learning Model for Short-Term Load Forecasting within Active Learning Framework, Energies, 2019, Vol. 12(20), 3809.

[3] Arakelyan, E.K., Sultanov, M.M., Boldyrev, I.A., Gorban, Y.A., Evseev, K.V. Analysis of the DCS historical data for estimation of input signal significance, Proceedings of the 3rd 2021 International Youth Conference on Radio Electronics, Electrical and Power Engineering, REEPE 2021, 2021, 9388069

[4] Arakelyan E.K. Application of machine learning methods for optimizing technical and economic performance of generating systems / Arakelyan E.K., Boldyrev I.A., Evseev K.V., Gorban Yu. A. // IOP Conference Series: Materials Science and Engineering, №1035, 2021