Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
Short term Markov corrector for building load forecasting system – Concept and case study of day-ahead load forecasting under the impact of the COVID-19 pandemic

Van Hoa Nguyen *, Yvon Besanger

Univ. Grenoble Alpes, CNRS, Grenoble INP (Institute of Engineering Univ. Grenoble Alpes), G2Elab, 38000 Grenoble, France

Abstract

In this paper, we present the concept and formulation of a short-term Markov corrector to an underlying day-ahead building load forecasting model. The models and the correctors are then integrated to the building supervision, control and data acquisition system to automate the self-updating and retraining processes. The proposed Markov corrector is experimentally proven to significantly improve the reactivity of the forecasting models with respect to untaught variations. Developed in a discrete manner over a continuous forecasting model, the corrector also helps to capture better the consumption peaks during the activity days. A proof-of-concept is demonstrated via the case study of the GreenER building, where the impact of the Markov correctors to the performance of the existing day-ahead load forecasting system (based on Prophet model) was analyzed during the 2021/2022 winter, under the influences of the Omicron wave of the COVID-19 pandemic.

© 2022 Elsevier B.V. All rights reserved.

1. Introduction

For over a century, the electrical grid has transported energy from the upstream production to the downstream distribution and consumption. This unidirectional architecture is nowadays fundamentally changed as more and more distributed renewable energy resources (DRES) are integrated to the downstream part, injecting the energy back to the grid. Essentially good for the environment, this new bidirectional grid introduces however a lot of new challenges to the grid operators: intermittency of the renewable sources leading to voltage and frequency fluctuation and potential grid congestion [1,2]. On the consumption side, we acknowledge a strong and continuous increase in terms of power demand, but also in terms of energy efficiency and conjunction with other types of energy (i.e. multi-energy system) [3].

Accurate and consistent load forecasting (LF) is essential for the good functionality of the electrical grid [4]. Long term LF enables precise strategies of generation scheduling and mitigating potential network congestion [4]. Medium term LF is necessary for planning and maintenance of electrical network [5]. Finally, short term LF (STLF), along with DRES production forecasting, is critical to balance the energy production/consumption and to determine the optimal usage of DRES for the advanced control of microgrid and for self consumption [6,7]. Challenging and dependent on the quality and amount of available data, LF is extensively treated using a wide range of advanced algorithms including regression methods (e.g. auto-regressive moving average (ARMA), auto-regressive integrated moving average (ARIMA)) [8,9], machine learning and deep learning (e.g. Support Vector Machine (SVM), Fuzzy Logic (FL), Genetic Algorithms (GA), Artificial Neural Network (ANN), etc.) [10,4,11,12]). An extensive review on different techniques of forecasting is presented in [13] and the different aspects as well as the positions in the following applications are discussed in [14].

In this paper, we focus only in the distribution scale and in particular the problem of building LF. At this level, building consumption is an important factor and accurate building LF is required for both the operator (e.g. demand response, advanced control, increasing DRES penetration) and the customer (e.g. improving self-sufficiency, consumption scheduling). Unlike forecasting at large scale, at building level, the human factor influences strongly the consumption behaviours [15]. In [16], we demonstrated how unprecedented event (e.g. CoVID-19 pandemic) can change the pattern of building consumption and how it impacts the performance of the building LF algorithm. It is therefore necessary to enable the continuous learning and self-updating features for the building LF system. In [17], an adaptive online learning...
method was proposed for building LF. However, the procedure is not yet fully automatic. In [16], we proposed the approach of connecting the building LF algorithm to its Supervision, Control And Data Acquisition (SCADA) system. In the proposed architecture, the self-updating process and the pre-correction of data (fixing time mismatch, missing points and outliers, etc.) is fully automatic. It was demonstrated in [16] that the forecasting performance is improved over time and the system is capable to adapt to new changes in users’ behaviour. The proposed system presented nevertheless several drawbacks. Based on the Prophet model [18], the machine learning module suffers from a “heavy inertia” of the historical data (i.e. takes around one month to adapt to new changes in behaviours). The model is continuous and sometimes fails to capture the consumption peak and the strong variation due to human factors. Besides, same model is used for different activity periods (e.g. weekday and weekend).

In this paper, we propose a short term corrector based on Markov chain to improve the accuracy and the reactivity of the developed system in [16]. The short term corrector aims to help the load forecasting system to better capture the peak and to adapt faster to the disruptive changes in consumption patterns or to randomness of human activities. The short time corrector can be maintained in different profiles to better capture the different features of different days/human patterns, etc.

In the following Section 2, we present the machine learning algorithm used for the LF module. The Markov corrector is presented in Section 3, along with its integration to a self-updating building load forecasting system. The case study of the GreenER building is presented in Section 4, where we analyze the impacts of the Markov corrector to the performance of the Machine learning algorithm. The paper is concluded in Section 5.

2. Self-updating building load forecasting system

We recall in this section the structure of the self-updating building load forecasting system proposed in [16]. The system is composed of three modules:

1. the machine learning module: responsible for training the models and making the forecast.
2. the self-updating module: responsible for real-time data exchange with the SCADA server and other applications and for triggering the retraining of the prediction models.
3. the self-evaluation module: based on the three criteria MAE, RMSE and MAPE.

In this paper, we focus on the machine learning module in proposed system in [16]. The module uses the Prophet model[18] fit with Limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm [19] to capture the strong seasonal effects of the building power demand. The Prophet model is formulated with the General Additive Model (GAM) ([20]):

\[ y(t) = g(t) + s(t) + h(t) + \epsilon(t) \]  

(1)

where:

- \( g(t) \) is the trend function (i.e. non-periodic changes):

\[ g(t) = \frac{C(t)}{1 + e^{-k(t-(m+a+t)^\gamma)}} \]  

(2)

with \( C(t) \) is the carrying capacity (the installed capacity of the local substation in our case study of building load forecasting), \( k \) the growth rate, and \( m \) an offset parameter. \( a \) and \( \gamma \) are respectively the vectors of rate and offset adjustments and \( a \) is the vector of adjustment accumulation counter.

- \( s(t) \) is the seasonality function (i.e. periodic changes modeled with flexible Fourier series):

\[ s(t) = \sum_{n=1}^{N} \left( a_{n}\cos\left(\frac{2\pi nt}{p}\right) + b_{n}\sin\left(\frac{2\pi nt}{p}\right) \right) \]  

(3)

where \( n \) is the order of the Fourier components, \( N \) is the truncating order and \( p \) is the period (yearly, weekly).

- \( h(t) \) is the independent holidays impacts:

\[ h(t) = [1(t \in D_1), \ldots, 1(t \in D_i)]^\kappa \]  

(4)

where \( D_i \) is the set of past and future dates for the holiday \( i \) and \( \kappa_i \) is the corresponding change in the forecast.

- \( \epsilon(t) \) represents other changes.

While the Prophet model was employed to minimize the necessity of human intervention, the proposed system can work well with a wide range of ML models (e.g. Artificial Neural Network (ANN), Long-short-term-Memory (LSTM), etc.) [21]. In the case of non-residential building, no technique has yet achieved significantly better performance than the others [22]. In the comparison in [7], state-of-the-art machine learning (ARIMA, Support Vector Regression (SVR-10-rbf, SVR-10-linear), Random Forest-9–200, eXtreme Gradient Boosting (XGBoost-5–100, and XGBoost-7–100)) and deep learning (Recurrent Neural Network (RNN-3–400, RNN-4–200), LSTM-3–200, LSTM-3–300, LSTM-4–400, GRU-3–100, and Gated Recurrent Units (GRU-4–300)) models got MAPE from 21.6% to 41.9% for short terms load forecasting for a building scale problem (evaluation on 1 h and 1 day data). In our case study in [16], the Prophet model achieved an average daily MAPE of 20.68%, 22.28% and 23.07% respectively for the models fit with preCovid, Covid and Mixed data – with the range of variation from 13% to 31% in September 2021 (i.e. one month duration). The performance can then be validated with respect to the other ML and DL algorithms for building load forecasting. Moreover, the proposed system proved its performance not only on a fixed dataset but also under the strong variation of real consumption (i.e. CoVID-19 context).

Below the threshold of 15–20% of MAPE error, it becomes more and more difficult to improve the accuracy of building STLF capacity based purely on time series predictors. This is not the accuracy limit of the forecasting models but is related more to the feature of the particular dataset (i.e. the building power demand). The residual error is mainly due to human and irregular activities in the building and is hard to reduced without getting more inputs on the building activities. On bigger scale (e.g. distribution or national load forecasting), these variations compensate themselves and the same tools can achieve better accuracy [15] (best known nowadays around 3–5% for national scale [23]).

At a building scale, besides the sudden weather fluctuations or load demand variations, unprecedented events (e.g. the COVID-19 pandemic) may result in a disruptive behaviour change and thus a strong deviation between the historical data set and the actual power demand. Using the self-updating mechanism proposed in [16], the model can automatically learn of the new scenario over time and prevent itself from being outdated. While it has been demonstrated that the proposed system can adapt to new data, the adaptation process is strongly influenced by the seasonality features of the building power demand.

We propose, in the next section, the use of a short term corrector based on Markov chain model to detect anomalies and disruptive changes in the consumption pattern of the building and to correct the relative errors. The output of the Prophet model is then introduced to the corrector, who tries to identify possible anomalies and changes using the relative errors of the base prediction and the actual power demand, before trying to compensate them.
3. Integration of a Markov corrector to the building load forecasting system

3.1. Load forecasting with Markov chain

Popular in state machine problems or for modeling random process, Markov chain [24] is based on the stochastic model in which the probability of each state depending only on its precedent state, regardless to the state before it. Using Markov chain as the modeling tool, some works have been done in the literature for the prediction of time series signals, particularly in short-term load forecasting, as a single model [25–27] or in combination with other models [28,29]. The use of Markov chain suffers from some drawbacks:

- The state space (i.e. distribution domain) has to be well-defined which means that the forecasting signal can not go beyond of that zone. This can be eventually coped with by defining the extremities covering the inf and sup of the distribution domain. However, in the case that the prediction falls into these states, the prediction is only meaningful for binary (fail/pass) problems and it can not provide much more useful information on the exact prediction value.
- As the model is based on a state system, it is necessary to adapt to time series forecasting. The adaptation approaches are mainly based on decomposing the definition domain of the signal into various states.
- It is hard to deduce the features and their contributions to the signal behaviour inside a Markov chain model.
- The accuracy of MK based predictors can only be improved by increasing the resolution of the decomposition, which increases exponentially the number of required computation for the transition matrix.

In our humble opinion, Markov chain may not be the best fit for building load forecasting or for time series forecasting in general, due to the above reasons. It is however a very powerful tool to complement/improve or to rectify the forecast of another predictor, thanks to its capability to model the pattern of errors between the predicted and the real values. In that vision, we propose to improve the machine learning model presented in Section 2 with a Markov corrector.

3.2. A Markov corrector for the Prophet model

We consider the power demand of the building represented by the time series:

\[ Y = \{ y_1, y_2, \ldots, y_n \} \]

(5)

The forecasted power demand fit by the main predictor is represented as:

\[ \hat{Y} = \{ \hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n \} \]

(6)

The series of relative prediction error is then defined as:

\[ E = \{ e_i = \frac{\hat{y}_i - y_i}{y_i} | i = 1, \ldots, n \} \]

(7)

We can then build the following error matrices:

\[ E^d = \begin{bmatrix} e^d_{h_1} & e^d_{h_2} & \ldots & e^d_{h_n} \\ e^d_{s_1} & e^d_{s_2} & \ldots & e^d_{s_n} \\ \vdots & \vdots & \ddots & \vdots \\ e^d_{m_1} & e^d_{m_2} & \ldots & e^d_{m_n} \end{bmatrix} \]

(8)

where \( d \) the number of days in the dataset and \( H = \{ h_1, h_2, \ldots, h_m \} \) consists of the daily points of sampling/measurement (e.g. \( H = \{ 00:00, 01:30, \ldots, 23:25 \} \)). In general, the common practice is to regularly distribute \( H \) over the day, e.g. 24 hourly samples per day. However, it is totally possible to add more frequent and unevenly distributed measures to a certain period of the day where the power demand varies more (e.g. peak time) and less frequent measures to more stable period (e.g. during the night), according to the desired requirements of the final application (e.g. load management).

To better identify the different features and characteristics of the building activities, we recommend separating this error series into two different datasets for weekdays (00:00 Monday to 23:59 Friday) and weekends (00:00 Saturday to 23:59 Sunday). In this way, we remove the transition of prediction performance pattern from weekday to weekend (and vice versa), out of the problem. This separation can then be generalized as “on activity” and “off activity” periods (e.g. a building may be closed on Wednesday and open on Sunday), or eventually into more individual profiles (e.g. “holiday”, “summer”), according to the real practice. Idem, it is possible to use two different sets and different distributions of \( H_{WD} \) for weekdays and \( H_{WK} \) for weekends.

\[
E_{WD} = \begin{bmatrix} e^d_{h_1} & e^d_{h_2} & \ldots & e^d_{h_m} \\ \vdots & \vdots & \ddots & \vdots \\ e^d_{s_1} & e^d_{s_2} & \ldots & e^d_{s_n} \\ \vdots & \vdots & \ddots & \vdots \\ e^d_{m_1} & e^d_{m_2} & \ldots & e^d_{m_n} \end{bmatrix}
\]

(9)

\[
E_{WK} = \begin{bmatrix} e^d_{h_1} & e^d_{h_2} & \ldots & e^d_{h_m} \\ \vdots & \vdots & \ddots & \vdots \\ e^d_{s_1} & e^d_{s_2} & \ldots & e^d_{s_n} \\ \vdots & \vdots & \ddots & \vdots \\ e^d_{m_1} & e^d_{m_2} & \ldots & e^d_{m_n} \end{bmatrix}
\]

Since the same procedure is applied to both datasets, in the following, we present only the general case and simplify the notation by omitting the WD and WK indices.

From (8), the rows \( E^d = [e^d_{h_1}, e^d_{h_2}, \ldots, e^d_{h_m}] \) represents the forecasting errors of \( h \) day and the columns \( E_j = [e^d_{h_1}, e^d_{h_2}, \ldots, e^d_{h_m}]^T \) represents the forecasting errors at the \( h \) sampling point for everyday in the dataset (e.g. same moment of the day, at 08:00 or at 11:34, etc.).

We decompose the series using the discrete and countable set \( \Theta_s \) of \( s \) states:

\[
\Theta_s = \left\{ \psi_1 : \varepsilon \in \left[ \inf \varepsilon, \psi_2 \right], \psi_2 : \varepsilon \in \left( \psi_1, \psi_2 \right], \psi_1 : \varepsilon \in \left( \inf \varepsilon, \sup \varepsilon \right] \right\}
\]

(10)

The choice of \( s \) is conditioned by the desired precision for the Markov corrector and the calculation complexity. We obtain the corresponding matrix or forecasting error states as:

\[
E^d = \begin{bmatrix} \theta^d_{h_1} & \theta^d_{h_2} & \ldots & \theta^d_{h_m} \\ \vdots & \vdots & \ddots & \vdots \\ \theta^d_{s_1} & \theta^d_{s_2} & \ldots & \theta^d_{s_n} \\ \vdots & \vdots & \ddots & \vdots \\ \theta^d_{m_1} & \theta^d_{m_2} & \ldots & \theta^d_{m_n} \end{bmatrix}
\]

(11)

In order to identify the forecasting performance and to predict the next possible forecasting errors at the moment \( h \in H \), we construct a Markov chain from \( E_j^d \):

\[
E_j^d = [\psi_{h_1}^d, \psi_{h_2}^d, \ldots, \psi_{h_m}^d]^T
\]

(12)
where \( \theta_1^d, \theta_2^d, \ldots, \theta_N^d \) are \( d \) observations taking values from the discrete states in the countable set \( \Theta \), and \( l \) is the “looking back frame” truncating \( \varepsilon_l \) on the last \( l \) observations.

For \( E_d \) at day \( d \), the transition probability \( P_{h_\ell \rightarrow h_f} \) from state \( h_\ell \) to state \( h_f \) is calculated as:

\[
P_{h_\ell \rightarrow h_f} = \frac{N_{h_\ell \rightarrow h_f}}{N_{h_\ell}}
\]

\( \forall (h_\ell, h_f) \in \Theta \); where \( N_{h_\ell \rightarrow h_f} \) is the number of occurrences of the transition from \( h_\ell \) to\( h_f \) and \( N_{h_\ell} \) is the total transitions originated from a state \( h_\ell \) in \( E_d \).

The Markov transition matrix for \( E_d \) can then be formulated as:

\[
\mathbb{P}^d = \begin{bmatrix}
P_{h_1 \rightarrow h_1} & P_{h_1 \rightarrow h_2} & \cdots & P_{h_1 \rightarrow h_N} \\
P_{h_2 \rightarrow h_1} & P_{h_2 \rightarrow h_2} & \cdots & P_{h_2 \rightarrow h_N} \\
\vdots & \vdots & \ddots & \vdots \\
P_{h_N \rightarrow h_1} & P_{h_N \rightarrow h_2} & \cdots & P_{h_N \rightarrow h_N}
\end{bmatrix}
\]

\( \mathbb{P}^d \)

Accordingly, we can calculate the transition matrix for every \( h_i \in \mathcal{H} \). These transition matrices represent the probability of how the accuracy of the forecasting model will behave given the historical error of the previous forecast.

Assuming at day \( d \), the forecasting errors are

\[
E^d = \{ \theta_1^d, \theta_2^d, \ldots, \theta_N^d \}
\]

and the main predictor provides the forecast

\[
\hat{Y}^{d+1} = \{ \hat{y}_{h_1}^{d+1}, \hat{y}_{h_2}^{d+1}, \ldots, \hat{y}_{h_N}^{d+1} \}
\]

\( \forall \theta_h^d \in E^d \), the most probable transition from \( \theta_h^d \) is calculated as:

\[
\hat{\theta}_h^{d+1} = \left\{ \theta \in \Theta : P_{\theta_h^d \rightarrow \theta} = \max_{h \in \Theta} \left( P_{\theta_h^d \rightarrow h} \right) \right\}
\]

In case that several transitions share equal probability (due to the same number of occurrences), we can set \( \hat{\theta}_h^{d+1} \) as the most recent observation in \( E^d \).

The predicted states of forecasting error are then:

\[
E^{d+1} = \{ \hat{\theta}_1^{d+1}, \hat{\theta}_2^{d+1}, \ldots, \hat{\theta}_N^{d+1} \}
\]

We can simply assume that the interval is normally distributed and the predicted error will be likely to fall into the mean of the interval. If \( \hat{\theta}_h^{d+1} \) takes the interval \( \theta_h \in \Theta_h \), then the estimated forecasting error at \( h \) is then the mean value:

\[
\hat{e}_h^{d+1} = \hat{\theta}_h
\]

We can then apply a correction to the initial forecast:

\[
\forall h \in \mathcal{H}, Y_{MK}^{d+1} = \frac{\hat{y}_h^{d+1}}{1 + \hat{e}_h^{d+1}}
\]

The corrected forecast for the next day is then:

\[
Y_{MK}^{d+1} = \{ Y_{MK}^{d+1}, Y_{MK}^{d+1}, \ldots, Y_{MK}^{d+1} \}
\]

Generally speaking, the proposed corrector works on the difference between the old behaviour (characterized by the base model) and the new behaviour (actual consumption data). The accuracy between the predicted values and the actual values can be characterized by the dominant pattern, represented by \( \mathcal{P}^d \), in this “looking back frame”. The choice of \( l \) is important for the corrector to decide if a particular change is a singular event or a behavioral shift and to propose the corrections to the models. The pattern is not considered as “changed” just after a few steps (days) and it is considered “changed” as it becomes the dominant in this looking back frame. The choice of \( l \) should therefore be long enough so that it does not become a “naive” model (i.e. copy from the previous day – sensible to short term anomalies) but should not be too long either to be able to reduce/eliminate the influence of the long term historical effects on the process.

The Markov corrector, as its name suggests, is built on an underlying predictor (Prophet model in this case). It contributes to reduce the impact of disruptive and random changes in human activities in the building but it does not replace the base predictor, which encompasses the other features.

3.3. Integration to a self-updating load forecasting system

In general, the integration of the Markov corrector to the self-updating process of the building load forecasting system [16] can be summarized as:

**Algorithm 1**

Procedure **SELF-UPDATING PROCEDURE** (Retraining)

Query the SCADA system for new measurements

Integrate the new data to the dataset

If “Warm-Start” (A Warm-start indicates that the model is retrained with previously trained model as input. A Black-start indicates that the model is retrained from scratch.)

Mode is ON then

load the old model as initial parameters.

else

Fit the model to the new dataset

Make prediction for the next forecasting horizon

Feed back the prediction to the corresponding services in the SCADA system.

Export the new model

Discretize and update Markov error data for the last forecasting period.

if Last forecasting period = “Weekday” then

Input the newly calculated errors to the weekday Markov error dataset.

else

Input the newly calculated errors to the weekend Markov error dataset.

end if

if Next forecasting period = “Weekday” then

Import from the weekday Markov error dataset.

else

Import the errors from the weekend Markov error dataset

end if

Construct the individual hourly Markov chain for errors at the same “hourly” moments during the day.

Calculate the new Markov Transition Matrix

From previous states (i.e. last period errors) and the transition matrix, calculate the most probable prediction errors for next period.

Calculate the estimated “Markov corrected” signal for hourly forecast.

Combine the hourly forecast to obtain the final day-ahead load forecasting profile.

Feed back the Markov corrected prediction to the corresponding services (e.g. SCADA).
where the steps from 10 to 24 are for the integration of new Markov corrector to the existing prophet-based machine learning models.

3.4. Generalized to other forecasting models

As indicated in [16], the Prophet model was chosen for its good capability to deal with seasonality. However, the machine learning module (cf. Section 2) can also employ other predictors. The MK corrector is not limited to the Prophet model but can be well applied to the employed model. It can prove to be useful when the building activities is subjected to a lot of human interference. In some cases, it is necessary to maintain several different prediction models for different profiles of the building (e.g. normal days, busy days, low activity season, etc.). Using multiple Markov profiles in these cases is simpler than maintaining and retraining multiple predictors.

As the Markov corrector can improve the reactivity of the forecasting model by adjusting the looking back frame, it is useful to quickly adapt the system with “heavy inertia”, with strong dependence on historical data without a lot of actual measurement.

It is recommended to maintain the underlying predictor independently to the addition of the corrector (i.e. separated datasets and processes) to avoid possible conflicts of resources and disturbances inside the system.

In the next section, we present the practical implementation of the proposed Markov corrector to the self-updating load forecasting system of the GreenER building. The accuracy and adaptation reactivity of the new predictor is compared to the ones without Markov corrector.

4. Case-Study: GreenER building load forecasting

4.1. Case-study description

The considered building is the 23,000 m² mixed education and research center Green-ER5 (Fig. 1), in Grenoble, France.

As presented in [16], the cloud server SCInterop provides day-ahead load forecasting for the Green-ER building via Software-as-a-service. The server is connected to the building SCADA system via an authenticated Simple Object Access Protocol (SOAP) Application Programming Interface (API) to update the model daily. Three forecasting models were developed with the power demand data from the pre-Covid period (01/01/2016 to the first national lockdown in France 3), Covid period (first lockdown onward) and Mixed (01/01/2016 onward) to analyse the impact of COVID-19 pandemic and the effect of continuously updating the models.

The analysis in [16] showed that the considered LF problem is influenced by multiple strong seasonality (daily, weekly schedule during school year) and is subjected to trend evolution (number of enrollment, installation). The main disturbance to the prediction is the variation of human activities.

The new Markov correctors with looking back horizon \( l = 30 \) days are added to the existing forecasting models and their performance is assessed mainly during the winter 2021–2022 where the activities were recovering.

Three metrics Mean Absolute Error (MAE), Root-Mean-Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) are used to evaluate the forecasting performance of the models with respect to the real power demand in a rolling basis (i.e. last 24 h).

4.2. Markov corrector impact analysis

We analyze in this part the impact of the added Markov corrector to the performance of the built forecasting models.

4.2.1. Consumption variation and its influence to the load forecasting system

Fig. 2 presents the difference of power demand between the period Oct 2021 – Mar2022 compared to the previous year historical data used in the model training (i.e. Oct 2020 – Mar 2021). A strong variation of power demand is acknowledged during the December 2021 – February 2022 period. The accuracy of the three pre-built models suffered strongly from this divergence of actual data versus the historical data, which is foreseeable. This strong variation, while can be credited to the intermittency of activities in the historical period (Fig. 2) is also partly related to the human factor – i.e. the pandemic evolution.

As clearly shown in the Fig. 2, a correlation can be observed between the number of daily confirmed Covid-19 case in France [30], its 7 days average and the accuracy of the pre-built models. The “low accuracy” period corresponds to the peak of the Omicron variant wave in France (i.e. Dec 2021 – Feb 2022). The accuracy of the models are stabilized as the number of cases is reduced in Mar 2022 before going up again in Apr 2022 as the number of cases re-increases. The daily number of cases at France level, while can not be interpreted directly as the situation in the building, can be used to relatively indicate how the building activities are affected (e.g. number of presence). During this period, among the three existing models the preCovid suffered the most as its dataset was fixed. The Covid model is the least influenced. From being the worst performing model before Nov 2021, the Covid model became the most “accurate” among the three pre-built models from Nov 2021 to Feb 2022 (Fig. 2). It shows that the power demand in this period exhibited more similarity to the historical data in the Covid period.

As in Fig. 3, after the aforementioned strong variation, the accuracy of the three models returns to around 20%. This improvement is largely influenced by the match between last year and this year power demand (Fig. 2) as well as the infection peak being over and the building activities became more regular. The self-updating process contributed only a small part to this improvement. Indeed, the preCovid model also performed better in March 2022 without any data update. Using a linear fit to the MAPE evolution of the three models, we can see the general dropping tendency of the MAPE of the preCovid Model (Fig. 4) while the Covid model accuracy tends to keep its level before the variation (Fig. 5). In fact, as the Covid model was not trained with the historical data in the pre-Covid period so it keeps on learning as new power demand is added. These added measurements are not enough yet for the model to improve its accuracy. On the other hand, the Mixed

\[
\text{MAE} = \frac{1}{N} \sum_{j=1}^{N} |P_{\text{actual}} - P_{\text{forecasted}}| \quad (22)
\]

\[
\text{RMSE} = \sqrt{\frac{1}{N} \sum_{j=1}^{N} (P_{\text{actual}} - P_{\text{forecasted}})^2} \quad (23)
\]

\[
\text{MAPE} = \frac{1}{N} \sum_{j=1}^{N} \frac{|P_{\text{actual}} - P_{\text{forecasted}}|}{P_{\text{actual}}} \cdot 100 \quad (24)
\]

where \( P_{\text{actual}} \) and \( P_{\text{forecasted}} \) are the actual and predicted power demands, respectively; and \( N = 24 \) is the number of predictions in the last forecasting horizon (24 h). We focus mainly in the MAPE criterion with the evaluation metrics in Table 1.

Same as the Prophet-based models configuration [16], the Markov-corrected models are also integrated to the SCADA system and the update process is fully automatic.
model with data from both periods performed slightly better than the preCovid model (Fig. 6). During April 2022, even though the number of Covid-19 cases started to raise again, the accuracy of the COVID model dropped while the others stay stably around 16–20%. This contrary to the previous period can be explained by both the relaxation of restrictions imposed by the government (e.g. removal of vaccine pass on 14 Mar 2022, finishing with

| MAPE Value | Evaluated quality | Suggested actions |
|------------|-------------------|------------------|
| 0–10%      | Excellent         | No               |
| 10–20%     | Good              | No               |
| 20–40%     | Fair              | Tweak parameters |
| > 40%      | Bad               | Notify the operator |

Table 1: Performance evaluation metrics and associated actions.

Fig. 1. GreenER building.

Fig. 2. From the top: Difference of power demand between the two periods: Oct 2021 – Mar 2022 and Oct 2020 – Mar 2021 – Its influence to the accuracy of the forecasting models – Correlation between the number of confirmed COVID-19 cases in France and the reduction of accuracy.
Fig. 3. Forecasting error compared to three months ago – without Markov corrector.

Fig. 4. Evaluation of performance improvement for preCovid model with the Markov corrector.

Fig. 5. Evaluation of performance improvement for Covid model with the Markov corrector.
work-from-home recommendation, etc.) and the distort from the last year data (during the pre-vaccination period of the pandemic) in the Covid model (Fig. 5).

Quantitatively, we can estimate roughly the improvement over time by comparing the error at the beginning and the end of the fitted trend over the six month period presented in Figs. 4–6.

$$\Delta_{\text{MAPE}} = \frac{\text{MAPE}_\text{last} - \text{MAPE}_\text{first}}{\text{MAPE}_\text{first}}$$

$$\|\Delta_{\text{MAPE}}\| = \frac{\text{MAPE}_\text{last} - \text{MAPE}_\text{first}}{\text{MAPE}_\text{first}}$$

While the Covid model’s accuracy is degraded for the aforementioned reasons, we acknowledge a strong improvement on all the other five models (Table 2).

The biggest performance gain came from the preCovid model as the building activities return to the normal rhythm. However, the impact of the Markov corrector can be easily seen as the corrected Covid model achieved 29.49% improvement despite the 96.15% degradation from the underlying Covid model, due to the dataset becoming irrelevant.

4.2.2. Impact of the Markov corrector to the performance

With the added Markov correctors, a clear improvement can be observed on all the three models (Fig. 2). While the Markov-corrected models performed slightly better (2–3% improvement on MAPE – i.e. 10% relative performance improvement) than the ones without corrector in “normal days” where the power demand matched historical data; they show very good reactivity in “strong variation” period. The Markov-corrected models needed only one or two iterations to get the accuracy to the range of 15–20%, which gives it much better average error in the long run. An observation over a three months window in Fig. 7 shows that the accuracy gradually improved; which is confirmed by the linear fit of MAPE in Figs. 4–6. The weekly ranges of variation of MAPE of the Markov-corrected models are also smaller than the non-corrected models – signifying the better consistency of the model performance (Figs. 4–6).

Moreover, as observed in Figs. 8–10, the Markov-corrected models captured better the peak of power demand during the weekday. In fact, unlike the pre-built models based on the GAM formulation, which is subjected to the continuity of the component functions, the Markov correctors work independently among different hours of day. They are therefore better adapted to the stiff rise of power demand during peak hours. This is an interesting property of the proposed approach.

In terms of accuracy improvement that the Markov corrector can bring, Table 3 presents the forecasting accuracy during the last 8 months. The average errors of the corrected models (preCovid, Covid and Mixed, in that order) are 18.99%, 22.96% and 21.9%, compared to 33.99%, 35.79% and 35.05% of the non-corrected Prophet models, i.e., $\|\Delta_{\text{MAPE}}\|$ of 44.13%, 43.50% and 44.93% improvement of accuracy respectively.

The non-corrected models are not that bad. For instance, they can achieve as low as sub 10% or eventually sub 5% of accuracy for some certain days (Table 3). If we investigate the 15 Feb to 15 Mar period (the transition period between the first and the second Omicron waves), they achieved comparable accuracy (sometimes even better) as the corrected models (Table 4). However, it took them a long time to adapt to the changes in behaviors and therefore they suffered from bad accuracy for a big part of the Omicron wave, leading to a low overall score.

We present an example for the adaptability of the corrected models, during the week of “Lundi de Paques” holiday (Fig. 11). The model predicted correctly the drop of power demand during the holiday (i.e. Monday 18th April 2022) and it precipitated a normal day on the next day. However, a lot of people profited the holiday to take a week off, leading to an abnormal drop in the activity. The model corrected itself and got good forecast profiles from Thursday onward (i.e. 2 days adaptation period).
Selectively, when the power demand is regular and does not possess a lot of fluctuations, the corrected model can achieve very impressive forecast accuracy. For instance, on 27th April, the corrected Mixed model reached the daily average MAPE error of 3.68% (Fig. 12).

5. Conclusion

In this paper, we present the concept and the implementation of a short-term Markov corrector to a self-updating building load forecasting system employing the Prophet model. Fully integrated to the building SCADA, the whole system is maintained and retrained automatically. The proposed Markov corrector helps to improve the reactivity of the machine learning module used for providing power demand forecast. It was proven to perform better in case of disruptive changes of the consumption pattern (human interference) or when the underlying forecasting model (Prophet in this case) suffers from heavy “inertia” (strong seasonality) of the dataset.

Defined in a discrete manner, the proposed corrector helps to better capture the peak of the daily power demand. The capacity to maintain different profiles for different periods of activity (e.g. weekday, weekend) is also a plus to fit better to the changes of pattern between these periods.

A proof-of-concept is demonstrated via the case study of the GreenER building, where the impact of the Markov correctors to the performance of the existing LF system (continuously and automatically updated) was analyzed during the 2021/2022 winter, with disruptive variation of power demand with respect to the 2020/2021 winter, along with the strong impact of the Omicron wave of the COVID-19 pandemic to the building activity. The results of the PoC confirmed that the Markov corrector significantly improves the reactivity of the forecasting models with respect to untaught variations and helps to capture better the consumption peaks during the activity days. Besides the reactivity, for
Fig. 9. Real and Forecasted power demand using the Covid model in March 2022 – with and without using the Markov corrector.

Fig. 10. Real and Forecasted power demand using the Mixed model in March 2022 – with and without using the Markov corrector.

| Table 3                      | Accuracy of the models from 05 Nov 2021 to 05 Jun 2022. |
|-------------------------------|---------------------------------------------------------|
| Model/MAPE                   | min       | max       | avg       |
| preCovid Model               | 5.85%     | 112.73%   | 33.99%    |
| Corrected preCovid Model     | 4.08%     | 78.19%    | 18.99%    |
| Covid Model                  | 6.94%     | 85.28%    | 35.79%    |
| Corrected Covid Model        | 4.43%     | 80.08%    | 20.22%    |
| Mixed Model                  | 8.04%     | 102.58%   | 35.05%    |
| Corrected Mixed Model        | 3.68 %    | 79.83%    | 19.3%     |

| Table 4                      | Accuracy of the models from 15 Feb 2022 to 15 Mar 2022. |
|-------------------------------|---------------------------------------------------------|
| Model/MAPE                   | min       | max       | avg       |
| preCovid Model               | 9.12%     | 26.49%    | 18.58%    |
| Corrected preCovid Model     | 15.22%    | 29.18%    | 20.13%    |
| Covid Model                  | 10.76%    | 39.73%    | 23.84%    |
| Corrected Covid Model        | 15.89%    | 29.74%    | 20.3%     |
| Mixed Model                  | 8.04%     | 23.77%    | 17.9%     |
| Corrected Mixed Model        | 16.84 %   | 30.05%    | 20.16%    |
some selective days, the corrected models also achieved very good forecasting accuracy, eventually sub 5% of MAPE.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Van Hoa Nguyen reports financial support was provided by IRICE.

Acknowledgments

The authors would like to thank Antoine Labonne, Tiansi Laranjeira and Julien Bemberger for their technical supports during the implementation of the system. This work is supported by the IRICE SGInterop project.

References

[1] A. Ahmad Khan, M. Naem, M. Iqbal, S. Qaisar, A. Anpalagan, A compendium of optimization objectives, constraints, tools and algorithms for energy management in microgrids, Renew. Sustain. Energy Rev. 58 (May 2016) 1664–1683.
[2] V.H. Nguyen, Q.T. Tran, H. Buttin, M. Guemri, Implementation of a coordinated voltage control algorithm for a microgrid via scada-as-a-service approach, Electr. Eng. (Mar 2021).
[3] M. Russo, V. Bertsch, A looming revolution: implications of self-generation for the risk exposure of retailers, Energy Econ. 92 (Oct. 2020) 104570.
[4] M. Mustapha, M. Mustafa, S. Khalidi, I. Abubakar, H. Shareef, Classification of electricity load forecasting based on the factors influencing the load consumption and methods used: an-overview, in: 2015 IEEE Conference on Energy Conversion (CENC0N), IEEE, 2015, pp. 442–447.
[5] N. Amjadi, A. Daraeepour, Midterm demand prediction of electrical power systems using a new hybrid forecast technique, IEEE Trans. Power Syst. 26 (2) (2010) 755–765.
[6] M. Ghiasi, D.K. Zimbra, H. Saidane, Medium term system load forecasting with a dynamic artificial neural network model, Electr. Power Syst. Res. 76 (5) (2006) 302–316.
[7] A.M.N.C. Ribeiro, P.R.X. do Carmo, P.T. Endo, P. Rosati, T. Lynn, Short- and very short-term firm-level load forecasting for warehouses: a comparison of machine learning and deep learning models, Energies 15 (3) (2022).
[8] M. Luy, V. Ates, N. Barisci, H. Polat, E. Cam, Short-term fuzzy load forecasting model using genetic-fuzzy and ant colony-fuzzy knowledge base optimization, Appl. Sci. 8 (6) (2018) 864.
[9] H. Dagdougui, F. Bagheri, H. Le, L. Dessaint, Neural network model for short-term and very-short-term load forecasting in district buildings, Energy Build. 203 (2019) 109408.
A. Al Mamun, M. Sohel, N. Mohammad, M.S.H. Sunny, D.R. Dipta, E. Hossain, A comprehensive review of the load forecasting techniques using single and hybrid predictive models, IEEE Access 8 (2020), 134,911–134,939.

C. Cao, L. Wu, Support vector regression with fruit fly optimization algorithm for seasonal electricity consumption forecasting, Energy 115 (2016) 743–745.

A. Bala, N. Yadav, N. Hooda, D. Registrar, Implementation of artificial neural network for short term load forecasting, Curr. Trends Tech. Sci. 3 (4) (2014) 247–251.

T. Hong, P. Pinson, Y. Wang, R. Weron, D. Yang, H. Zareipour, Energy forecasting: a review and outlook, IEEE Open Access J. Power Energy 7 (2020) 376–388.

G. Cao, L. Wu, Support vector regression with fruit fly optimization algorithm for seasonal electricity consumption forecasting, Energy 115 (2016) 734–745.

A. Bala, N. Yadav, N. Hooda, D. Registrar, Implementation of artificial neural network for short term load forecasting, Curr. Trends Tech. Sci. 3 (4) (2014) 247–251.

T. Hong, P. Pinson, Y. Wang, R. Weron, D. Yang, H. Zareipour, Energy forecasting: a review and outlook, IEEE Open Access J. Power Energy 7 (2020) 376–388.

G. Gürses-Tran, A. Monti, Advances in time series forecasting development for power systems; operation with mlps, Forecasting 4(2) (2022) 501–524. [Online]. Available: https://www.mdpi.com/2571-9394/4/2/28.

A. Preumont, Markov Process, Springer, 1994.

V. Alvarez, S. Mazuelas, J.A. Lozano, Probabilistic load forecasting based on adaptive online learning, IEEE Trans. Power Syst. 36 (4) (Jul. 2021) 3668–3680.

J. Munkhammar, D. van der Meer, J. Widén, Very short term load forecasting of residential electricity consumption using the markov-chain mixture distribution (ncm) model, Appl. Energy 282 (2021) 116180.

A. Preumont, Markov Process, Springer, 1994.

H. Meidani, R. Ghanem, Multiscale markov models with random transitions for energy demand management, Energy Build. 61 (2013) 267–274.

J. Wang, J. Kang, Y. Sun, D. Liu, Load forecasting based on gm – markov chain model, in: 2010 Second Pacific-Asia Conference on Circuits, Communications and System, vol. 1, 2010, pp. 156–158.

H. Früh, D. Groß, K. Rudion, Short term load forecasting for individual consumers based on markov chains, in: 2019 Modern Electric Power Systems (MEPS), 2019, pp. 1–5.

A. Asrari, D. Javan, H. Javidi, M. Monfared, Application of gray-fuzzy-markov chain method for day-ahead electric load forecasting, Przeglad Elektrotechniczny, vol. 2012, 12 2012, pp. 228–237.

E. Dong, H. Du, L. Gardner, An interactive web-based dashboard to track covid-19 in real time, Lancet. Infect. Dis. 20 (5) (2020) 533–534.