Machine Learning based Load Prediction

Prof. Veena B[1*], S B Pallavi[2]

[1]*Assitant Professor, Department of Master of Computer Application, UBDTCE, Davangere*
[2]*S B Pallavi, PG Student, Department of Master of Computer Applications, UBDTCE, Davanagere*

ABSTRACT

Load forecasting is very important tool for energy suppliers and also for other participants in electric energy generation, transmission and distribution systems. It plays an important role in the power system planning and operation. Load forecasting has great impact on power system applications such as energy purchasing, energy generation and infrastructure development. Many mathematical methods are used for load forecasting. The load forecasting reduces capital investment on the equipments to be installed. In this project, the focus is on Short-Term Load Forecasting (STLF) which is an hourly load forecasting on next day. In our proposed project STLF will be carried out using one of the conventional methods, Multiple Linear Regression (MLR) and modern method, Artificial Neural Network (ANN). The ANN and MLR here use the data such as past load and weather information like humidity and temperatures. Once the Artificial Neural Network is trained for the past set of data, it can give prediction of future load. Finally, there is a comparison drawn between MLR and ANN on the basis of their Mean Average Percentage Errors (MAPE).

INTRODUCTION

Load forecasting is the method for prediction of electrical load. Load forecasting can be defined as the technique to estimate of how much electricity will be needed in the future. Load forecasting is an important tool for the energy management of electrical power system. Precise load forecasting helps the electric utility to make unit commitment decisions, reduce spinning reserve capacity and schedule device maintenance plan properly. For optimal power system operation, electrical generation must follow electrical load demand. The generation, transmission, and distribution utilities require some means to forecast the electrical load. So that they can utilize their electrical infrastructure efficiently, securely, and economically.

Generation utilities use electrical load forecasting techniques to schedule their generation resources to meet the future load demand. Transmission utilities use electric load forecasting techniques to optimize the power flow on the transmission network to reduce congestion and overloads.

Distribution utilities would not have much interest in short-term electric load forecasts, their distribution systems are predominantly radial with predictable maximum load demands. Thus, the distribution systems are sized conservatively and short-term load changes have little effect on the distribution system. Load forecasting can be categorized into three major divisions they are short-term load forecasting, medium- term load forecasting and long-term load forecasting. Short-term load forecasting is from one hour to one week which is used to supply necessary information for system management regarding day to day operations. Medium forecasting is usually from a week to a year, used to supply electric utility company management with prediction of future needs for expansion, equipment purchases, or staff hiring. Long-term forecasting is longer than a year, used for scheduling fuel supplies and unit maintenance.

Since STLF can be used to reduce operating cost, electric supplier will use forecasted load to control the quantity of running generator units. It is important to supplier because they can use the forecasted load to control the quantity of running of generators in operation. Thus STLF is very Important for electricity trading. Therefore it is necessary to establish high accuracy model for STLF. Here we have taken the traditional method called Multi Linear Regression and a modern method Artificial Neural Network to compare accuracy of results.

Literature Review

A plethora of approaches consisting of time series analysis, regression, smoothing techniques, artificial inteligence, artificial neural networks, machine learning, deep learning, reinforcement learning and various hybrid methods can make this area of research overwhelming. Some authors suggest that established models are better [7–9] presents evidence that complexity harms accuracy. Authors in [10] propose the Golden Rule to provide a unifying theory of forecasting, while others embed multiple algorithms to build hybrid methods combining characteristics of traditional statistics and machine learning. There is truth on both sides; some algorithms will work better or worse depending on historical data or applied period. The forecasting objectives are to minimize errors and improve economic activity: revenue, profit, and higher customer satisfaction. Low error forecasts are of no inherent value if ignored by the industry or otherwise not used to improve organizational performance. Forecasting competitions presented in [11,12] is one of the best ways to compare algorithms on reliable historical data and point out Appl. Sci. 2021, 11, 10126 3 of 18 the results. In multiple cases, recurrent neural networks (RNN) architecture stands out as a stable algorithm. The work done by [13] presents an extensive experimental study using seven popular DL architectures and found that LSTM is the most robust type of recurrent network, and while LSTM provides the best forecasting accuracy, convolutional neural networks (CNN) are more efficient and suffer less variability of results. In this paper, various RNN networks are applied for industrial load forecasting and analyzed to establish the best architecture for deep recurrent neural networks. In the article [14] authors point out that the difference between simple RNN to GRU and LSTM is that the number of parameters increases, a conclusion also presented in our article. For 24 h ahead forecasting commercial building data, authors concluded that the DNN model achieved worse results than the sequence to sequence RNN models. The authors in [15] present a simple recurrent neural network for the one-hour prediction of residential electric load. The model takes as inputs weather data as well as data related to electricity consumption. The percentage error calculated for a week test is 1.5% for the mean error and 4.6% for the maximum error. The difference between industrial load and residential usage is that the latter is highly dependent on weather data and daily patterns are more repetitive. In our article, exogenous variables such as temperature, humidity, and dew point are used in forecasting, because the industrial processes analyzed are influenced by these variables. Day-ahead forecasting of hourly large city load based on deep learning is studied by [16] with a novel flexible architecture that integrates multiple input features processed using different types of neural network components according to their specific characteristics. The authors have implemented multiple parallel CNN components with different filter sizes to introduce parallel structure into the DNN model instead of stacking DNN layers. The proposed architecture (MAPE: 1.405%) outperformed the CNN-LSTM (MAPE: 1.475%) and the DNN (MAPE: 1.665%). Another approach based on RNN and CNN is proposed by [17], consisting of convolutional layers and bidirectional LSTM and GRU recurrent layers to predict the next hour utility load. The results of experiments on two datasets (0.67% and 0.36% MAPE) demonstrate that the proposed model outperforms the conventional GRU and LSTM models. In this article, we found that the deep GRU network performs better than the combined GRU + LSTM network. A comprehensive comparison performed by authors in [18] concludes that RNNs require more resources than traditional models, but perform better. We reached similar findings in our article, the GRU unit is simpler than the LSTM unit, as well as faster in computations. The article presents that overall the LSTM performs better than GRU, which contradicts the results for short-term load forecasting presented in our article. The authors in [19] compare different variations of the LSTM algorithm and conclude that the longer the historical data available for training, the better the load forecasting accuracy would be. For building loads, the day-ahead forecasting errors show up to 45% improvement using RNNs (LSTM, LSTM with attention, BiLSTM, BiLSTM with attention) in comparison with other states of the art forecasting techniques

Materials and Methods

The forecasting methods are implemented in this paper use hourly data (Figure 2) from an industrial company active in the wood processing industry for an entire year (2019). The power supply for the factory is provided through twelve power transformers summing 12.6 MVA. The following technological processes, machinery, and equipment determine the electricity consumption forecasted in this article:

• Installations that serve the equipment for cutting and exhaust;

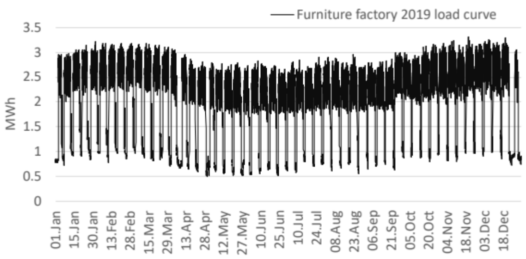
• Installations that serve the cooling system to ensure the necessary cold to keep in optimal conditions the substances used in the foaming process;

• Installations that serve the processing and cutting of sponges;

• The installations that serve the different subsections when making the mattresses;

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 07 | July - 2022 Impact Factor: 7.185 ISSN: 2582-3930

• Equipment used for making upholstery and assembling all subassemblies;



• Interior lighting installations located in the physical perimeter of all production halls;

• Robots for packing finished products

• Conveyors for the transport of products in the logistics warehouse;

• Specific facilities for food preparation in the canteen;

• other installations are specific to the universal processes which take place within this undertaking.

For the implementation Tensorflow [20] was used for deep learning applications. Keras [21] is a high-level API, open-source library for machine learning that works on top of Tensorflow. For the data preparation and visualization of the results, Scikit-learn [22], Numpy [23], and Seaborn [24] were used. The simulations computed on a PC Intel(R) Core(TM) i5-4690K CPU@3.5 GHz, RAM 16 GB, 64-bit operating system, x64-based processor. The industrial consumer analyzed is a furniture factory consisting of all the technological processes necessary to manufacture furniture starting from raw wood, mainly electric drives. The consumer energy needs are electricity and wood scraps. Production of heat and hot water relies on burning the remaining wood from the technological processes. The heating in the winter period for the office building and factory production facilities is achieved with electric heaters which influence the consumption in the winter period together with the lightning systems (work schedule is in three shifts). High electricity consumption is driven by large ventilated storage halls used for the thermal preparation of the raw wood. Correlation between electric load and outdoor temperature, dew point, and humidity is observed. Working/non-working days load patterns are not the same because factory planning is highly dependent on production quota. A Dickey–Fuller test [25] made for the yearly load time series points to the null hypothesis and the non-stationarity of the time series. Reliable linear dependencies between exogenous variable and consumption could not be establish and deep learning became an option to explore for nonlinear dependencies. From all the algorithms implemented in this article, variations of RNN (LSTM, GRU, GRULSTM), the GRU algorithm offered the best result for forecasting. Given this reason, we tried to analyze which is the best structure for the GRU for our particular problem.

Machine learning

There is a vast spectrum of terminology that tends to be confusing because of the interchangeability of utilization: artificial intelligence, machine learning, deep learning, artificial neural networks, or reinforcement learning. Machine learning is considered a subdomain of artificial intelligence [26]. Deep learning is a subdomain of machine learning, and neural networks are at the core of deep learning algorithms. The dissimilarity between a simple neural network and a deep learning algorithm is the number of neurons and structure of hidden layers (deep learning must have more than two hidden layers). ML techniques can be broadly grouped in two large sets—supervised and unsupervised. The methods related to the supervised learning paradigm classify objects in a pool using a set of known annotations/attributes/features. The unsupervised learning techniques.

form groups among the objects in a batch by identifying similarities and then use them for classifying the unknowns. Reinforcement learning is a behavioral algorithm similar to supervised learning, not using sample data for training but by trial and error. A sequence of successful outcomes will develop the best recommendation or policy for a given problem. DL models were developed to map a complex function between the last “n” hours (timesteps—also called lag) and predict how the time series can continue in the future, as presented in Figure 3. Most machine learning algorithms have hyperparameters; by setting the parameters, the ML algorithm can offer the desired results. The values of hyperparameters should not be calculated in the learning stage (because of the overfitting problem). To evaluate the generalization of the DL methods on the training data, we use a testing set of time series that the built network in the training stage did not experience prior. In our work we use deep recurrent neural networks (DRNN) and variations of the algorithm. RNN is a sequential data neural network processor because it has internal memory to update the state of each neuron in the network with the previous input. Because RNN train with backpropagation, this can fail because of vanishing gradient descent. Deep networks combine multiple layers into the architecture and provide more significant benefits.

Neural networks build functions by multiplying a weight matrix to the input vector, add bias, and then apply the activation function to obtain non-linearity in the output. To calculte the current state, we can use the following Formula (1). h(t) = f(h(t−1) , x(t) ; θ) = tanh Wh(t−1) + Ux(t) + b (1) In the equation above, the parameters θ include W, U, and b. The W and U are parameters representing weight matrices, and b is the bias vector. Hyperbolic tangent is the activation function tanh for the hidden state; other activation functions could be used. The output of the RNN cell is: o(t) = g(h(t) ; θ 0) = Vh(t) + c (2) where V and c denote the weight and bias, the parameters θ 0 of the output function g. Matrix V and vector c are multidimensional outputs. The same set of parameters is applied at each time step for every RNN-cell [27]. LSTM was developed to improve the vanishing or exploding gradient problem and has become one of the most popular RNN architectures to date and was introduced by [28]. GRUs were later introduced by [29] as a simpler alternative and have also become quite popular. We will use both architectures in the context of the vanishing or exploding gradient problem. Many variants of LSTM and GRUs exist in the literature, and even the default implementations in various deep learning frameworks often differ. Performance is often similar, but this can confuse when reproducing results. The study proposed by [30] ranked MLP first in terms of forecasting performance, better than Support Vector Regressi

Conclusion

A compromise is needed to find a practical solution to make electric load forecasting more accessible to the industry sector by implementing algorithms that learn directly from data with little human intervention. The novelty of the work is the proposed framework applied for industrial load curves, the analysis of the best architecture, and the scalability of the deep neural networks using a simple complexity index. The study compared the forecast performance for seven methods and tested various combinations for forecast variables and lag structures. Our test sample results across 1608 hourly values (15 October–20 December 2019) indicate consistently that: (i) deep recurrent neural networks are suitable for industrial load consumption; and (ii) the best model implemented for is GRU. The work highlights that increasing the number of hidden layers and neurons in each layer can negatively impact the performance of the DL algorithms.

References

1. Hernandez, L.; Baladron, C.; Aguiar, J.M.; Carro, B.; Sanchez-Esguevillas, A.J.; Lloret, J.; Massana, J. A Survey on Electric Power Demand Forecasting: Future Trends in Smart Grids, Microgrids and Smart Buildings. IEEE Commun. Surv. Tutor. 2014, 16, 1460–1495. [CrossRef]

2. Chambers, J.C.; Mullick, S.K.; Smith, D.D. How to Choose the Right Forecasting Technique; Magazine; Harvard University, Graduate School of Business Administration: Cambridge, MA, USA, 1971.

3. Armstrong, J.S. Selecting Forecasting Methods. Princ. Forec. Int. Ser. Oper. Res. Manag. Sci. 2001, 30, 365–386. [CrossRef]

4. Archer, B.H. Forecasting demand: Quantitative and intuitive techniques. Int. J. Tour. Manag. 1980, 1, 5–12. [CrossRef]

5. Feinberg, E.A.; Genethliou, D. Load Forecasting. In Applied Mathematics for Restructured Electric Power Systems: Optimization, Control, and Computational Intelligence; Chow, J.H., Wu, F.F., Momoh, J., Eds.; Springer: Boston, MA, USA, 2005; pp. 269–285. [CrossRef]

6. Nti, I.K.; Teimeh, M.; Nyarko-Boateng, O.; Adekoya, A.F. Electricity load forecasting: a systematic review. J. Electr. Syst. Inf. Technol. 2020, 7, 13. [CrossRef]

7. Gravesteijn, B.Y.; Nieboer, D.; Ercole, A.; Lingsma, H.F.; Nelson, D.; Van Calster, B.; Steyerberg, E.W. Machine learning algorithms performed no better than regression models for prognostication in traumatic brain injury. J. Clin. Epidemiol. 2020, 122, 95–107. [CrossRef]
8. Kuster, C.; Rezgui, Y.; Mourshed, M. Electrical load forecasting models: A critical systematic review. Sustain. Cities Soc. 2017, 35, 257–270. [CrossRef]

9. Green, K.C.; Armstrong, J.S. Simple versus complex forecasting: The evidence. J. Bus. Res. 2015, 68, 1678–1685.

10. Armstrong, J.; Green, K.; Graefe, A. Golden rule of forecasting: Be conservative. J. Bus. Res. 2015, 68, 1717–1731. [CrossRef]

11. Hyndman, R.J. A brief history of forecasting competitions. Int. J. Forec. 2020, 36, 7–14. [CrossRef] Appl. Sci. 2021, 11, 10126 17 of 18 12. Makridakis, S.; Spiliotis, E.; Assimakopoulos, V. The M4 Competition:

100,000 time series and 61 forecasting methods. Int. J. Forec. 2020, 36, 54–74. [CrossRef]

13. Lara-Benítez, P.; Carranza-García, M.; Riquelme, J.C. An Experimental Review on Deep Learning Architectures for Time Series Forecasting. Int. J. Neural Syst. 2021, 31, 2130001. [CrossRef]