Forecasting Electric Load by Aggregating Meteorological and History-based Deep Learning Modules

Masoud Bashari
IEMS Group Ltd, IBM Innovation Space
Toronto, Canada
masoud.bashari@iemsgroup.ca

Ashkan Rahimi-Kian
IEMS Group Ltd, IBM Innovation Space
Toronto, Canada
arkian@iemsgroup.ca

***Abstract*— Accurate day-ahead (or 24-hours ahead) electric load forecasting for power systems is crucial for system's optimal operations. In evolving smart distribution grids, the importance of precise electric load forecast in day-ahead is even more important for distributed energy management systems (DERMS) and demand response (DR) programs, which are used by the independent system operators (ISO) and power utilities (PU) for day-ahead system planning and optimal operations. This paper captures both dynamic members' interdependencies and the impact of meteorological factors on the load sequence. In this regard, Long Short-Term Memory (LSTM) is applied to use the historical load sequences to forecast the 24 hours ahead values of the system load. On the other hand, a Deep Feedforward Neural Network (DFNN) is applied to map the forecasted meteorological parameters to the upcoming 24-hourly values of the system load. Finally, based on the historical errors of these two engines, a Beta distribution generates a probabilistic weight for aggregating the forecasted values at each hour. The proposed forecasting model performs better than single engines based on Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Square Error (RMSE) metric when applied to day-ahead load forecasting for the city of Toronto.**

Keywords*— Deep Feedforward Neural Network (DFNN), Long Short-Term Memory (LSTM), Short-Term Load Forecasting, Local Distribution Company (LDC), Distribution System Operator (DSO), Independent Electricity System Operator (IESO), Hourly Ontario Electricity Price (HOEP).

I. INTRODUCTION

Moving towards smart energy grids, short-term load forecasting has become an increasingly important problem whose solutions can be categorized as: methods based on numerical weather prediction (NWP), methods based on statistics or machine learning, and hybrid approaches. The electric load forecasting problem is crucial for any power system planner and operator, because any gap (due to load forecast error) between supply and demand can cause large penalties (imposed by the ISO) and higher market clearing prices to meet the hourly loads by the LDC (or DSO). In fact, reducing the peak load and load variations is one of the demand response management (DRM) tasks, which could be done more efficiently with the aid of accurate short-term load forecasting (STLF) [1]. A hybrid load forecasting structure is proposed in this paper that can capture both dynamic interdependencies in the load sequence and the effect of meteorological factors on the electric load.

In recent years, researchers have applied various deep learning frameworks to forecast the power system load in short term. As an example, Long Short-Term Memory (LSTM) based architecture was applied to predict the residential loads in [2], where the authors achieved promising results; however, they disregarded the effect of meteorological parameters on the load. Moreover, Deep Belief Network (DBN) was applied in [3] to forecast the load in power distribution networks; however, the authors only considered the temperature as the influencing meteorological input and neglected the effects of other parameters on the electric load.

In several research works, the authors have tried to utilize deep learning algorithms for short-term load forecasting. In [1], the authors used Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) to generate an initial set of features; then, they fused these features to provide a final prediction of the electric load. In [4], the authors provided a two-stage framework for forecasting the electricity price. In the first stage, they predicted the load by using Gated Recurrent Unit (GRU) and Convolutional Neural Network (CNN) blocks, and then used the load forecast results as an input into another CNN engine for day-ahead electricity price forecasting. In [5], in order to extract the informative features and dynamics of the historical load sequence, the authors used Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), respectively to predict the load in one stream. In another parallel stream, the calendar and weather features were applied to predict the load via a Dense Neural Network layer. Finally, by Concatenation of the Dense layers, the two predicted load streams were fused. In another research work [6], the volatility components were extracted from the historical data and along with the historical meteorological data were fed into a Deep Belief Network (DBN) on one side; on the other side, the time series components were fed into a Nonlinear Auto Regressive (NAR) Dynamic Neural Network. The final predicted load streams were fused together. In [7], in order to predict the peak load more precisely, a parameter called Value-at-Risk (VaR) was computed using the Copula model. The applied Copula model was fitted based on electricity price and temperature. Furthermore, the authors used a Deep Belief Network (DBN) to forecast the complete load profile.

In [8], the authors improved the Quintile Regression Neural Network (QRNN) for stochastic electric load forecasting. The probabilistic load forecasting can be fed into power system simulation & optimization software tools, which can use random system parameters (such as load and price) for power system planning and optimal operation. To provide a more accurate probabilistic load forecasting, the authors of [9] proposed constrained quintile regression averaging (CQRA) method to combine interval forecasting engines including:

978-1-7281-5508-1/20/$31.00 ©2020 IEEE

Quintile Regression Neural Network (QRNN), Quintile Regression Random Forests (QRRF), and Quintile Regression Gradient Boosting (QRGB).

By reviewing the relevant literature, it is seen that the predicted meteorological parameters have not been efficiently used for electric load forecasting in the relevant literature. Furthermore, in hybrid load forecasting models, the aggregation module is complex and needs a sophisticated training procedure. In this paper, a hybrid algorithm is proposed, which feeds the forecasted weather parameters directly into the load forecast engine. Furthermore, our proposed hybrid forecasting model utilizes a simple aggregation mechanism, which provides competitive results in comparison with its counterparts in the literature.

This paper proposes a simple heuristic architecture for hybrid load forecasting. The proposed model is simple and easy to implement. Furthermore, it performs online and improves the forecasting accuracy in near real-time applications. The rest of this paper is organized as follows. Section II presents the basic formulation of Long Short-Term Memory (LSTM) and Deep Feedforward neural networks (DFNN). Section III provides the architecture of the proposed aggregation model. Section IV presents the simulation results with thorough analysis and finally, Section V concludes the paper.

II. Long Short Term Memory (LSTM) and Deep Feedforward Neural Networks (DFNN)

The proposed aggregation mechanism has three main components, which are Long Short-Term Memory (LSTM), Deep Feedforward Neural network (DFNN) and Beta distribution-based aggregation mechanism. In this section, the preliminaries of LSTM and DFNN are studied.

A) History based deep learning module based on Long Short-Term Memory (LSTM):

Long Short-Term Memory (LSTM) is a Recurrent Neural Network (RNN) in which the problem of cascade chain gradient during the learning process is handled. Hence, it can be easily used to predict sequences. Equations (1-5) provide the input-output relations of the Long Short-Term Memory (LSTM) cells [1].

$$i_t = \sigma(W_i \times [h_{t-1}, x_t] + b_i) \tag{1}$$

$$f_t = \sigma(W_f \times [h_{t-1}, x_t] + b_f) \tag{2}$$

$$C_t = f_t \times C_{t-1} + i_t \times (tanh(W_c \times [h_{t-1}, x_t] + b_c)) \tag{3}$$

$$o_t = \sigma(W_o \times [h_{t-1}, x_t] + b_o) \tag{4}$$

$$h_t = o_t \times tanh(C_t) \tag{5}$$

In equations (1-5), W_i, W_f, W_c, and W_o are weighting matrices; b_i , b_f, b_c, and b_o are bias vectors; i_t, f_t, and o_t are the outcomes of input, forget and output gates respectively. C_t is the cell state and h_t is the final output of the LSTM. Moreover, x_t is the input, where in this paper it represents the time series of the load. Using LSTM for electric load forecasting, we should decide about the number of applied LSTM cells, which depends on the number of hidden layers and the number of steps in chunking the electric load sequence. Furthermore, we should decide about the batch size while we train the network.

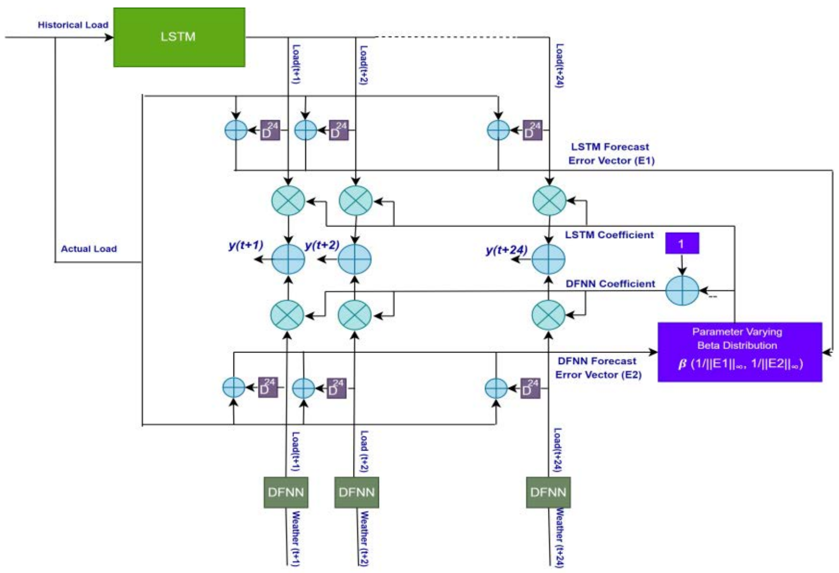

Fig. 1. The proposed architecture of combined weather-based and history-based deep learning neural networks for 24-hour load forecasting

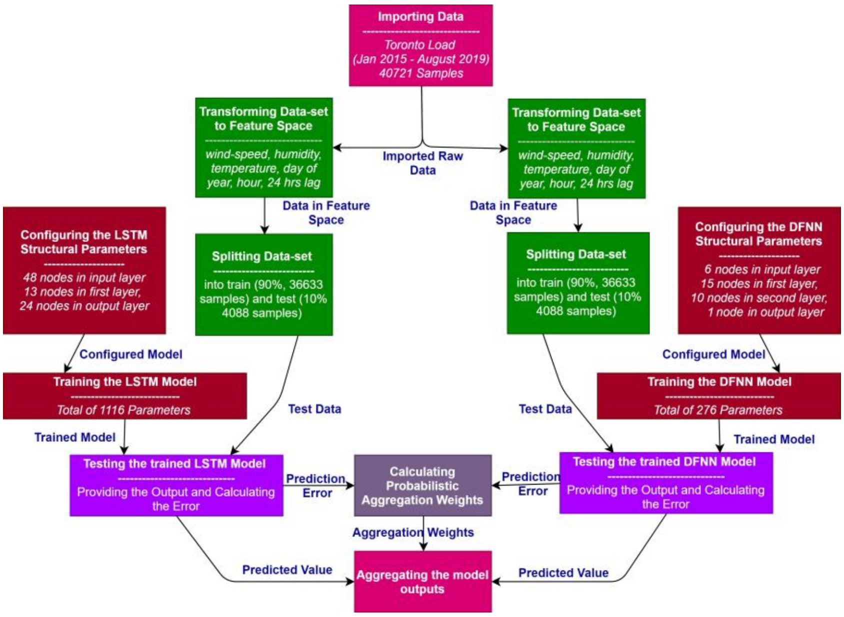

Fig. 2. The simulation workflow

B) Weather based deep learning module based on Deep Feedforward Neural Network (DFNN)

Usually, a Feedforward Neural Network (FNN) whose number of layers is more than three is called a Deep Feedforward Neural Network (DFNN). In this architecture we apply 24 DFNN modules for forecasting the load profile in the next 24 hours. For load prediction, each DFNN has 6 inputs, which are the forecasted temperature, forecasted wind speed, forecasted humidity, hour of the day at the time of forecast, day of the year at the time of forecast, and a binary value (0 for week day and 1 for weekend and holiday). Fore real time implementation, the forecasted meteorological values are obtained via Python APIs.

III. The Architecture of Aggregation Solution

The proposed architecture is presented in Figure 1. This architecture has three main components: 1) A time-series predictor based on Long Short-Term Memory (LSTM), 2) A series of Deep Feedforward Neural Networks (DFNN), which predicts the load using the forecasted meteorological data, and finally, 3) An aggregation module that provides a weighted average of the previous modules' errors. Here, the weights are probabilistic and generated by a Beta distribution whose expected value is proportional to the error of that module.

In the aggregation module, first, the forecasting error is calculated for each hour for both LSTM history-based and weather-based DFNN modules. The error vectors for LSTM and DFNN computational engines are called E1 and E2, respectively with 24 elements each. Since, all the load data are normalized, the amplitudes of all 24 error elements are less than 1. Hence, both $\frac{1}{||E2||_\infty}$ and $\frac{1}{||E1||_\infty}$ are more than one. While $||E1||_\infty$ increases, $\frac{1}{||E1||_\infty}$ decreases and the LSTM coefficient's probability density function will be skewed toward zero. Hence, the LSTM's coefficient will be more likely close to zero and the DFNN's coefficient will be more likely close to one.

IV. Simulation and Results

The proposed architecture is implemented within the IBM Watson Studio with Python 3.6 using the workflow presented in Figure 2. The Toronto's load data along with the weather data were collected from the IESO website. There are 40721 records in the dataset, which cover the period of January 2015 to August 2019. This data set is split into train dataset (90%, 36633 data samples) and test dataset (10%, 4088 data samples). The LSTM is configured with 48 nodes in the input layer, 13 nodes in the hidden layer and 24 nodes in the output layer. The 48 input nodes are used for feeding the load values of the last 48 hours to the network. On the other hand, the prediction of upcoming 24 hours is provided by 24 nodes in the output layer. For the DFNN module, there are 6 nodes in the input layer which are the forecasted temperature, forecasted wind speed, forecasted humidity, hour of the day at the time of forecast, day of the year at the time of forecast, and a binary value (0 for week day and 1 for weekend and holiday) at the time of forecast. Each DFNN has one output, which is the forecasted load at the desired time. Figure 3 presents the actual load profile vs. the 24-hour ahead load predictions for August 17, 2019 and Aug 25, 2019. To evaluate the overall performance of the forecasting engines, Table 1 compares the three models in the sense of Root Mean Square Error (RMSE), Normalized Root Mean Square Error (NRMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) for 170 days in the test dataset.

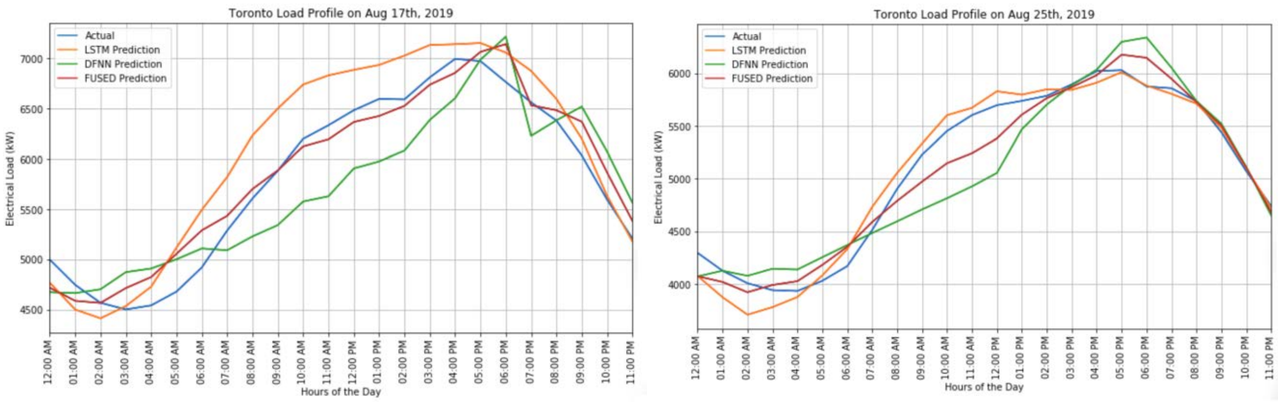

Fig. 3. Actual vs. predicted values for August 17, 2019 (Left) and August 25, 2019 (Right) Toronto load forecast

TABLE 1. COMPARISON BETWEEN THE HYBRID MODEL, LSTM AND DFNN BASED ON FOUR STANDARD PERFORMANCE INDICES FOR ELECTRIC LOAD

FORECAST METHOD	LSTM	DFNN	Hybrid Model
Root Mean Square Error (RMSE) [MW]	395.03	350.98	304.09
Normalized Root Mean Square Error (NRMSE)	0.0577	0.0513	0.0444
Mean Absolute Error (MAE) [MW]	334.9	279.05	250.21
Mean Absolute Percentage Error (MAPE) [%]	5.84%	4.77%	4.28%

Another important aspect about precise load forecasting for a given power network for the system operator (ISO, TSO or DSO) is the Energy Market Loss. If a DSO's day-ahead load forecast is "under-forecast", then it has to purchase the residual demand [{L_{real} (t) – $L_{forecast}$(t)} > 0] from the spot (or balancing market) at a higher (and more volatile) market price. On the other hand, if the DSO's day-ahead load forecast is "over-forecast", then it will purchase extra power [{L_{real} (t) – $L_{forecast}$(t)} < 0] from the day-ahead energy market ahead of time, and there is no guarantee that the ISO/TSO will buy it back from the DSO in real-time (i.e. via Demand Response Auction). So, the DSO's Energy Market Loss (or Risk) for any day may be computed as follows for both "under-forecast" and "over-forecast" errors:

$$DSO_{Loss}(day) = \sum_{t=1}^{24}\{Err_{uf}(t) \times LMP_{RT}(t) + Err_{of}(t) \times LMP_{DA}(t)\} \quad (6)$$

Where, Err_{uf}(t) is the "under-forecast" error [in MW] at hour *t*; Err_{of}(t) is the "over-forecast" error [in MW] at hour *t*; LMP_{RT}(t) is the real-time market clearing price [in \$/MWh]; LMP_{DA}(t) is the day-ahead market clearing price [in \$/MWh]; and DSO_{Loss}(day) is DSO's daily market loss [in \$].

To show the real market losses for a typical LDC (or DSO) in greater Toronto area (GTA), we evaluated the energy market loss [in \$] for August 17, 2019 load forecast considering the day-ahead and real-time IESO's HOEP prices. Table 2 shows the IESO day-ahead energy market loss for the typical LDC in GTA for August 17, 2019, and the average annual loss (in 2019) for the same LDC due to load forecast error imposed by the LSTM, DFNN and the proposed Hybrid Model methods.

TABLE 2. COMPARING THE LSTM, DFNN AND HYBRID MODEL LOAD FORECASTING METHODS BY MEANS OF ENERGY MARKET LOSS [\$M]

DSO Market Loss [\$M]	LSTM	DFNN	Hybrid Model
August 17, 2019	1.514	1.86	1.21
2019 Average	553	680	442

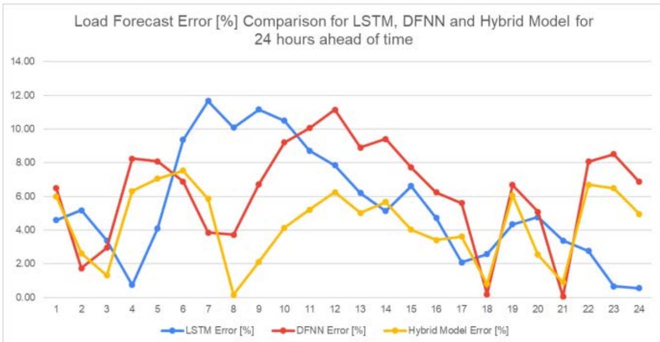

Fig. 4. Load forecast absolute hourly errors [%] for the LSTM, DFNN and Hybrid Models on August 17, 2019

The load forecast absolute hourly error percentages for the LSTM, DFNN and Hybrid Model on August 17, 2019 are shown in figure 4. These error values together with day-ahead and real-time HOEP prices were used to compute the DSO's energy market loss [in \$M] on August 17, 2019.

Table 2 shows that the Hybrid Model with 4.37 MAPE (for August 17, 2019) compared to LSTM (5.46 MAPE) and DFNN (6.35 MAPE) has the minimum amount of IESO's energy market loss (1.21 \$M) among the three tested methods.

V. CONCLUSION

Improving the accuracy of power networks' load forecasting is crucial for distribution network planning, optimal operations and minimum purchase of reserves (for load balancing, frequency and voltage regulation) to the Distribution System Operators (DSO). As the number of smart meters (AMI) is increasing within power distribution networks, the state-of-the-art deep neural networks (DNNs) can be utilized to provide more accurate load forecasts to the DSOs.

In this paper, a Beta distribution was applied to generate probabilistic aggregation weights for two forecasting engines: the first developed forecasting engine was Long Short-Term Memory (LSTM), which captures the dynamic members' interdependencies in the load sequences. The second developed forecasting engine was Deep Feedforward Neural Network (DFNN), which is applied to map the forecasted meteorological parameters to the upcoming 24 values of electric load. Applying the developed hybrid load forecasting models (fused LSTM & DFNN) to Toronto real electric load data, it was shown that the fused LSTM & DFNN model improved the load forecast accuracy vs. either LSTM or DFNN model considerably by means of MAE, MAPE, and RMSE error evaluation indices.

To show the importance of precise network load forecasting for a typical DSO (in Toronto), we calculated and presented the expected energy market loss [in $M] on August 17, 2019 and the expected average annual loss (for 2019) [in $M] based on the "over-forecast" and "under-forecast" errors and IESO's day-ahead and real-time HOEP prices for the LSTM, DFNN and Hybrid Model. Table 2 showed that the Hybrid Model with 4.37 MAPE (for August 17, 2019) compared to LSTM (5.46 MAPE) and DFNN (6.35 MAPE) had the minimum energy market loss (1.21 $M) among the three tested methods.

Following this work, we are going to extend the proposed aggregation mechanism to provide quintile based probabilistic forecasting of the electric load to increase the accuracy.

REFERENCES

[1]. C. Tian, J. Ma, C. Zhang, P. Zhan, "A Deep Neural Network Model for Short-Term Load Forecast Based on Long Short-Term Memory Network and Convolutional Neural Network," *Energies 2018*, vol. 11, no. 3493, pp. 1-13, 2018.

[2]. W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, 2019, 841-851.

[3]. S. Motepe, A. N. Hasan, B. Twala and R. Stopforth, "Power Distribution Networks Load Forecasting Using Deep Belief Networks: The South African Case," *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, Amman, Jordan, 2019, pp. 507-512.

[4]. M. Afrasiabi, M. Mohammadi, M. Rastegar and A. Kargarian, "Probabilistic deep neural network price forecasting based on residential load and wind speed predictions," in *IET Renewable Power Generation*, vol. 13, no. 11, pp. 1840-1848, 2019.

[5]. W. He, "Load Forecasting via Deep Neural Networks," *Information Technology and Quantitative Management (ITQM 2017), Procedia Computer Science 122 (2017)*, pp. 308–314, 2017.

[6]. P. Yi, Z. Jianyong, Y. Yun, Z. Rui, Z. Cheng and S. Tian, "An Electricity Load Forecasting Approach Combining DBN-Based Deep Neural Network and NAR Model for the Integrated Energy Systems," *2019 IEEE International Conference on Big Data and Smart Computing (BigComp)*, Kyoto, Japan, 2019, pp. 1-4.

[7]. T. Ouyang, Y. He, H. Li, Z. Sun, and S. Baek, "Modeling and Forecasting Short-Term Power Load with Copula Model and Deep Belief Network," *IEEE Transactions on Emerging Topics in computational Intelligence*, vol. 3, no. 2, 2019, 127 - 136

[8]. W. Zhang, H. Quan, and D. Srinivasan, "An Improved Quantile Regression Neural Network for Probabilistic Load Forecasting," I*EEE Transactions on Smart Grid*, vol. 10, no. 4, 2019

[9]. Y. Wang, N. Zhang, Y. Tan, T. Hong, D. S. Kirschen, and C. Kang, "Combining Probabilistic Load Forecasts," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 3664-3674, 2019.