Short-Term Electrical Load Prediction for Future Generation Using Hybrid Deep Learning Model

Conference Paper · February 2022

DOI: 10.1109/ICAEEE54957.2022.9836359

3 authors:

s.M. Anowarul Haque Sonet
Mymensingh Engineering College

Gobinda Chandra Sarker
University of Dhaka

Kazi Md Sadat
Mymensingh Engineering College

Short-Term Electrical Load Prediction for Future Generation Using Hybrid Deep Learning Model

1st S. M. Anowarul Haque
Dept. of EEE
Mymensingh Engineering College
Mymensingh, Bangladesh
engr.sonet2017@gmail.com

2nd Gobinda Chandra Sarker
Dept. of EEE
Mymensingh Engineering College
Mymensingh, Bangladesh
gcsarker001@gmail.com

3rd Kazi Md Sadat
Dept. of EEE
Mymensingh Engineering College
Mymensingh, Bangladesh
kmdsadat07@gmail.com

***Abstract*—Power generation is increasing worldwide every year to cope with ever-increasing energy demand. Therefore, a significant necessity exists for forecasting the load demand to manage and increase electricity production capacity. Short-term load forecasting (STLF) using artificial neural network has become one of the most efficient and widely popular methods. This paper proposes a hybrid network of Long Short-Term Memory (LSTM) network and Convolutional Neural Network (CNN) to predict demand for seven days into the future. The proposed CNN-LSTM method is compared with various deep learning techniques such as vanilla neural network and gated recurrent unit (GRU).Power Grid Company of Bangladesh (PGCB) has the responsibility of reliable power transmission all over the country. Each model is trained and tested on multivariate historical data collected from the daily report section of PGCB website for the Mymensingh Division in Bangladesh. Various input features such as temperature, peak generation at evening, maximum generation, month and the season of the year are used to aid the prediction. It is found that the proposed CNN-LSTM method outperforms the other models with a MAPE error rate of 2.8992%, which is less than the MAPE error of 5.5554% for demand estimation of seven days used by PGCB.**

***Index Terms*—load forecasting, STLF, LSTM, CNN-LSTM, neural network, ANN**

I. Introduction

Accurately load forecasting is necessary for the operation, planning, and distribution of electrical energy. The ability to forecast load demand is essential in developing power station infrastructures. Overproduction of power at specific-time results in a waste of energy while underproduction is unsatisfactory at the customer level. Therefore, it is crucial to predict the necessary load in advance to supply power accurately and save energy.

Load forecasting is a method that predicts the load needed at a particular time. Load forecasting can be broadly classified into three categories [1] : Long-term, Mid-term, and, Short-term Load forecasting. Long-term load forecasting has a time duration of from a few months to a year. Mid-term load forecasting lasts from a week to a month. Short-term load forecasting (STLF) lasts from several hours to one week. The load demand depends on several environmental factors [2], [3] such as temperature, humidity, wind speed.

The load forecasting research is mostly region-based [4] in Bangladesh. There are different methods applicable for load-forecasting such as statistical methods, machine learning, and deep learning methods. Studies have found, deep learning methods to be more accurate in load forecasting than other methods. Different deep learning models such as LSTM, CNN, CNN-LSTM models have shown excellent performances. Therefore, for the better maintenance of the power system, more research is essential. In our study, we have worked on STLF using different deep learning methods that are good at handling complex time-series data. We have collected six years of historical data on daily load demand from the PGCB website for Mymensingh division. Our study is intended to predict the load seven days ahead. We compared our studies to the predicted Load of PGCB. Our work aims to the following contributions.

- We have investigated the performance of several deep learning techniques such as LSTM, GRU, CNN-LSTM on our dataset and compared them to determine the best method for STLF.
- PGCB does not follow any fruitful method for short-term load forecasting. Our study, intended to apply the most effective deep learning models for the purpose of the best load prediction results. It can significantly improve the load-forecasting capabilities of the company.
- Our studies can be helpful for other power-supplying companies for load-forecasting and facilitate research.

II. Recent Works

Civilization is advancing while technology is playing a pivotal role. With the drastic improvement of technology, there is an ever-increasing demand for power generation. Power generation is expensive and often needs to depend on natural resources. Therefore, accurate load forecasting (LF) is needed to manage the limited resources properly for planning and producing electrical energy at a specific time interval, which in turn also reduces overall cost for the production. Heinemann et al. [4] discovered a relation between environmental factors and demand for power in 1966. Lijesen et al. [5] further studied LF and proposed some statistical methods. Linear and non-linear models for load forecasting have been a topic of interest for research of load forecasting in the recent years. Some linear and non-linear methods were researched using Artificial Neural networks (ANN) [6]. Several other methods

978-1-6654-6944-9/22/$31.00 ©2022 IEEE

such as fuzzy logic, Generic Algorithm (GA) [7] have shown effective results for load forecasting. In recent times Recurrent Neural Network (RNN) has been proven to be an effective method [8] for load forecasting. Long Short-Term Memory (LSTM) [9] is proven to be a successful method for load forecasting that can work without external variables to forecast load demands. This method is good at learning seasonal variances. Convolutional-Neural Network with Long-Short Term Memory (CNN-LSTM) method is a hybrid neural network model. Some recent studies have found this model effective, especially in hourly-based [10] Short-Term load forecasting.

Some studies have been done in Bangladesh in region basis. For STLF utilizing Artificial Neural Network (ANN) with a Particle Swarm Optimization (PSO) model [11] and LSTM models [12], [13] have shown promising results. The methods used by the PGCB to forecast load could be improved using deep learning methods. We approached the study trying to find the performances of different deep learning models. We also introduced CNN-LSTM model which has shown promising aspects in STLF.

III. Forecasting Model

Our proposed model architecture is CNN-LSTM which is briefly discussed in the following sections.

A. Long Short-Term Memory Network

A recurrent neural network (RNN) is effective for sequence data analysis like our dataset. This is due to the fact that, traditional feed forward network has no memory while recurrent neural network loops through the sequence components and maintaining a state of previous information. In this work, we have used two variants of RNN called long short-term memory (LSTM) and gated recurrent unit (GRU). Normal RNN suffers from the gradient vanishing [14] issue when learning long-term dependencies. LSTM solves this problem by using a memory cell state which may carry information across many timesteps without changing information. The basic structure of LSTM block for a single timestep is given in Fig. 1. LSTM interacts with the cell state [15] through three gates that determine which sequence component to forget, update or pass through. We assume $(x_1, x_2, x_3, ..., x_t)$ is the input sequence where x_t is a vector at timestep t. The node outputs of a LSTM block are calculated as follows [16].

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

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{2}$$

$$\tilde{C} = \tanh(W_c[h_{t-1}, x_t] + b_c) \tag{3}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C} \tag{4}$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t * \tanh(C_t) \tag{6}$$

Where f_t, i_t and o_t indicates the forget, input and output gate. σ symbolizes sigmoid activation which converts the gate output between 0 and 1 to determine how much information of cell state need to be deleted, modified or preserved. W_f, W_i, W_c and W_o are the weight matrix learned during the training of the model and control gates behavior. The equations contain bias values which are represented by b_f, b_i, b_c and b_o.

B. Convolutional Neural Network

Convolutional neural network (CNN) can recognize local patterns in the input sequence. Due to this reason, CNN has been extensively applied in the time series forecasting [17] in recent years. CNN extracts local patches and uses convolution operation to produce output feature map or filters. These filters incorporate specific aspects such as position of the pattern in the input. The operation is computed using Eq. 7 the following formulae to get output feature map in two-dimensional axis.

$$s = (I * K)(i, j) = \sum_m \sum_n I(m, n)K(i - m, j - n) \tag{7}$$

Where I and K are the input sequence and kernel respectively. The indexes of the output feature map is denoted by i and j. The maximum pooling operation involves down sampling the feature map by sliding a small window through the filters and outputting the maximum value. Pooling is required to contain more information about the totality of the input and to reduce overfitting. For our historical data we have utilized one dimensional convolutional layer. The abstract representation for 1D-convolutional layer [18] is shown in Fig. 2. Here 'A', 'B' and 'F' represents two convolutional layers and one feed

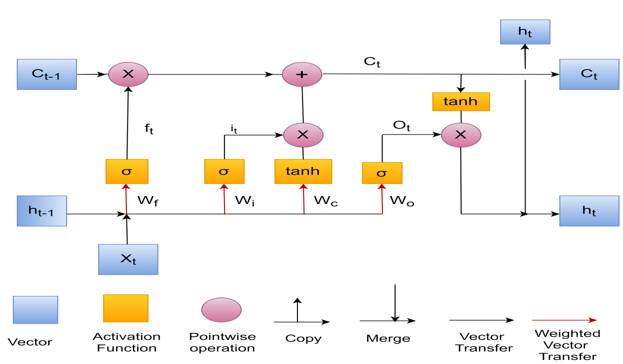

Fig. 1. LSTM architecture for a single timestep with gates

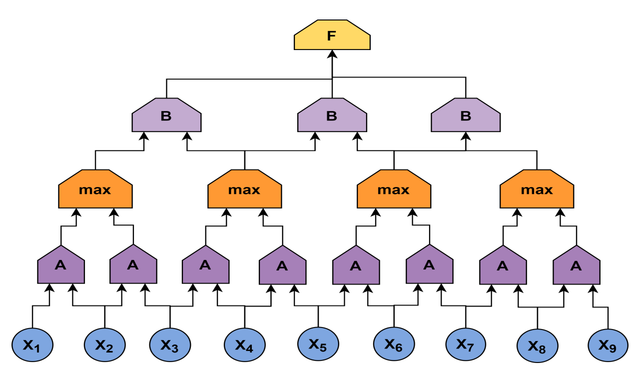

Fig. 2. Abstract process representation of 1D convolutional NN.

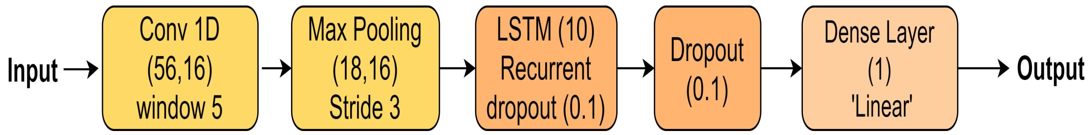

Fig. 3. Block diagram of proposed LSTM-CNN model.

forward layer respectively. $(X_0, X_1, X_2......X_n)$ denotes the input sequence.

C. Hybrid CNN-LSTM network

Our proposed CNN-LSTM model is implemented by combining convolutional neural network module and long short term memory module as a hybrid framework. The CNN-LSTM network uses CNN as a preprocessing step before feeding into the LSTM layer to deliver high precision forecasting. The purpose of using such hybrid model is, this combination works better with longer sequences. CNN module works as a feature extractor by capturing local trends in the input sequences. CNN part consists of a convolutional layer followed by a max-pooling layer. The convolutional layer is implemented with rectified linear unit (ReLU) activation function. Activation function improves the capabilities of the model to learn highly complex structure. Maxpooling layer downsamples the input sequence before feeding into the LSTM layer thus reducing computational load. LSTM utilizes internal cell states and learn the dependencies in the sequence. Recurrent dropout is utilized with the LSTM block to mitigate overfitting. Dropout is a process that randomly deactivates some neuron during model training. The architecture used for this hybrid model is shown in Fig. 3. The final output is taken from a dense layer.

IV. Methodology

A. Data Preparation

We have prepared our dataset from the daily report section of Power Grid Company of Bangladesh (PGCB) website [19]. In this paper, seven features; maximum generation (MW), actual peak day (MW), actual generation at Evening (MW), temperature, weekday, month, season are used to predict the demand Since demand depends on these features. We have found 53.94%, 59.43%, 82.59% and 22.24% correlation between demand and maximum generation, actual peak day, actual generation evening and temperature respectively. Our dataset contains the daily energy generation and demand of the Mymensingh division in Bangladesh from January 2014 to December 2019. Features such as month, weekday, season are one hot encoded. 80% of the dataset is used for training, the rest is used for evaluation. Data processing involves standardization or normalization. In our experiment, we have found that, normalized data performs better than the standardized dataset. We determined the minimum and maximum value in each category and calculated the normalized value for each data using Eq. 8 and 9. Where, for column X, x represents a single data point, x_{min} and x_{max} indicate the minimum and maximum value of column X respectively. The value of (a,b) represents the numerical range we want to convert our data. We have converted our data to the range (0,1). The normalized data is prepared with a batch size of 32 to be the input of our model. We have used python 3 and Keras Tensorflow library to preprocess, train and validate our model in Google Colab cloud platform.

$$s = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{8}$$

$$x_{scaled} = s(b - a) + a \tag{9}$$

B. Model Building And Training

In this research work, we have applied four neural network architectures on our dataset. These architectures include traditional vanilla neural network, long short term memory network (LSTM), gated recurrent unit (GRU) and a hybrid network of LSTM and convolutional neural network. All the models are optimized to predict seven days into the future while looking back 60 days of past data. The vanilla NN is implemented using two hidden feed forward layer of 16 units and an output dense layer. After training, the output of this model acts as a baseline for our experiments. The proposed LSTM and GRU network both uses two LSTM and GRU layers respectively of 10 nodes followed by a dense layer for output. The LSTM-CNN network uses a convolutional neural-network as a preprocessing step before feeding into

TABLE I
HYPERPARAMETERS OF DIFFERENT MODELS

Hyperparameters	Vanilla NN	LSTM	GRU	CNN-LSTM
Filter window size	-	-	-	5
Pooling	-	-	-	max
No. of hidden layers	2	2	2	1 Conv1D + 1 LSTM
No. of units	16/16	10/10	10/10	16 +10
Recurrent dropout	-	0.1	0.1	0.1
Lookback	60	60	60	60
delay	7	7	7	7
Optimizer	Adam	Adam	Adam	Adam
Learning rate	0.001	0.001	0.001	0.001

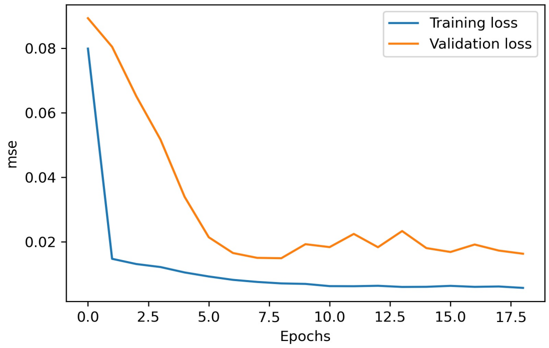

Fig. 5. Training and Validation loss Vs Epoch curve for LSTM-CNN model using mse error rate

the LSTM layer. For training sequence data like our dataset, we have used one dimensional convnet which extracts local 1D patches of window size 5, followed by a max pooling layer of stride 3. The convnet and pooling layer turns the input sequence into down-sampled high level features. These features are fed to the LSTM model of 10 nodes. A recurrent dropout of 0.1 is applied in the LSTM layer. The output of each network is taken from a final dense layer of one node with linear activation. The training and validation loss curve found during the training of our proposed model is shown in Fig. 5. The hyperparameters used in different models are listed in Table I. The models are trained with 'EarlyStopping' with a batch size of 32 and the best weight parameters are stored. The hyperparameters are optimized by experimenting with the models several times to achieve the best performance.

V. Result Analysis

The proposed model as well as the other models in this paper is trained to predict load demand seven days into the future. The models are trained on historical data from January 2014 to December 2019 (six years) at daily basis and evaluated on dataset from the first week of January 2020 in terms of mean absolute percentage error (MAPE). Table III shows MAPE rate on test data for different models. The visual representation of the error rate found by different models is shown in Fig. 4, where it is seen that LSTM and GRU models perform very close to each other achieving MAPE error rate of 3.4096% and 3.4798% respectively, while CNN-LSTM model presents the lowest value of error. CNN-LSTM model is found to give the best performance with an MAPE error rate of 2.8992% outperforming the baseline vanilla NN model by 61.05% and PGCB estimated demand error by 47.81%.The prediction curve for CNN-LSTM model on train and test data is shown in Fig. 6 and 7. These figures verify that our proposed models can follow the demand trend. The Actual and predicted demand by CNN-LSTM model for the first week of January 2020 is given in Table II. The estimated demand forecasted by PGCB is provided as well. It can be observed that, the

TABLE II
Actual and predicted demand for seven days using CNN-LSTM model

Day	Actual Demand (MW)	Demand Predicted by the proposed Model (MW)	PGCB estimated By PGCB (MW)
01/01/20	653	623.31	625
02/01/20	640	621.75	605
03/01/20	636	636.64	595
04/01/20	608	635.11	588
05/01/20	659	632.53	594
06/01/20	660	642.49	599
07/01/20	651	640.14	620

TABLE III
MAPE error rate of different models.

Model	MAPE (%)
PGCB estimated Demand Error	5.5554
Feed Forward NN	7.4432
GRU	3.4798
LSTM	3.4096
LSTM-CNN	2.8992

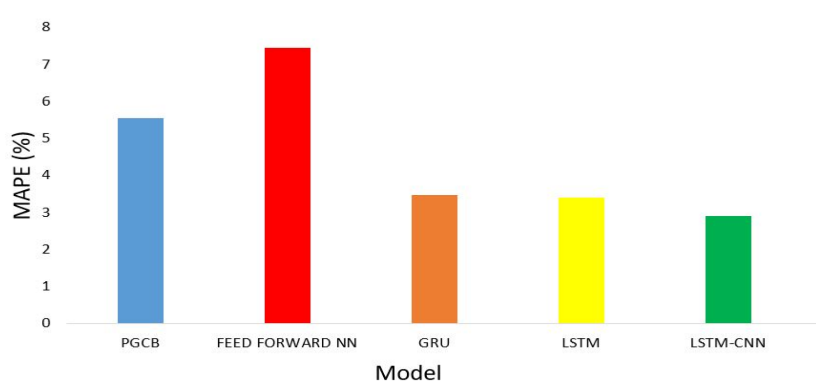

Fig. 4. Visual representation of MAPE error found by different models

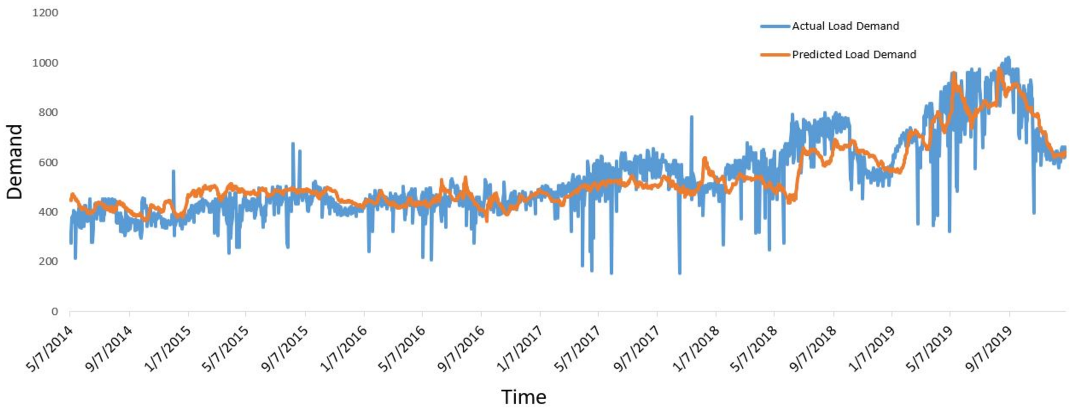

Fig. 6. Actual Vs Predicted load curve of LSTM-CNN model on past data

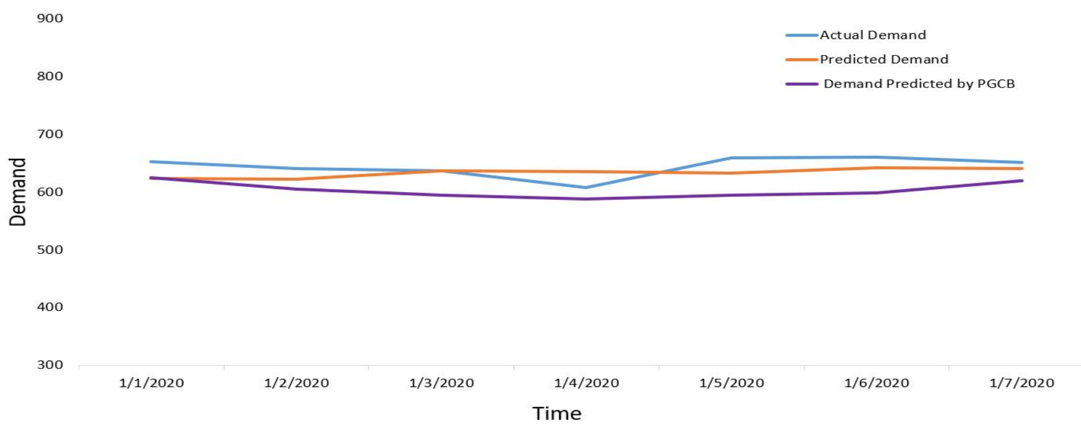

Fig. 7. Actual Vs Predicted load curve of LSTM-CNN model for future seven days

predicted output of our proposed model is much closer to the actual demand.

VI. Conclusion

This paper focused on the implementation of advanced deep learning models for STLF. Four different deep learning models were implemented for the collected historical data of actual load demand and predicted load from the PGCB, using multivariate features consisting of maximum generation, generation at evening, peak day, temperature, month, weekday, and season for the Mymensingh division in Bangladesh. CNN-LSTM model is proposed in this paper, aiming at improving the efficiency of STLF for its superior efficiency over other models. This study concludes the CNN-LSTM model as the best model achieving a MAPE error rate of 2.89%. LSTM and GRU models also showed satisfactory results with MAPE errors of 3.41% and 3.47%, which are less than the MAPE error of 5.47% by the PGCB. Our proposed model has shown 47.81% improvement than the PGCB estimated demand error, thus proving its superiority. The proposed model can be utilized by PGCB as well as other power supply companies to enhance the load-forecasting accuracy reducing the power generation cost. Neural networks learn more precisely when more features are available in the training process. In future works, features such as humidity, holidays can be added to facilitate the training process of deep learning models for better load predictions. Other evaluation technique such as forward chaining cross validation can be explored. The performances of models such as GRU-CNN, transformers, N-BEATS can also be studied and investigated in the future for more precise load forecasting.

References

[1] A. K. Singh, S. Khatoon, M. Muazzam, D. Chaturvedi *et al.*, "Load forecasting techniques and methodologies: A review," in *2012 2nd International Conference on Power, Control and Embedded Systems*. IEEE, 2012, pp. 1–10.

[2] M. Sobhani, T. Hong, and C. Martin, "Temperature anomaly detection for electric load forecasting," *International Journal of Forecasting*, vol. 36, no. 2, pp. 324–333, 2020.

[3] J. Xie and T. Hong, "Wind speed for load forecasting models," *Sustainability*, vol. 9, no. 5, p. 795, 2017.

[4] G. Heinemann, D. Nordmian, and E. Plant, "The relationship between summer weather and summer loads-a regression analysis," *IEEE Transactions on Power Apparatus and Systems*, no. 11, pp. 1144–1154, 1966.

[5] D. Lijesen and J. Rosing, "Adaptive forecasting of hourly loads based on load measurements and weather information," *IEEE Transactions on Power Apparatus and Systems*, no. 4, pp. 1757–1767, 1971.
[6] D. C. Park, M. El-Sharkawi, R. Marks, L. Atlas, and M. Damborg, "Electric load forecasting using an artificial neural network," *IEEE transactions on Power Systems*, vol. 6, no. 2, pp. 442–449, 1991.
[7] C. Li, "A fuzzy theory-based machine learning method for workdays and weekends short-term load forecasting," *Energy and Buildings*, vol. 245, p. 111072, 2021.
[8] L. Sehovac and K. Grolinger, "Deep learning for load forecasting: Sequence to sequence recurrent neural networks with attention," *IEEE Access*, vol. 8, pp. 36 411–36 426, 2020.
[9] B.-S. Kwon, R.-J. Park, and K.-B. Song, "Short-term load forecasting based on deep neural networks using lstm layer," *Journal of Electrical Engineering & Technology*, vol. 15, no. 4, pp. 1501–1509, 2020.
[10] H. Kuang, Q. Guo, S. Li, and H. Zhong, "Short-term power load forecasting method in rural areas based on cnn-lstm," in *2021 IEEE 4th International Electrical and Energy Conference (CIEEC)*. IEEE, 2021, pp. 1–5.
[11] S. A. Haque and M. A. Islam, "Artificial neural network-based short-term load forecasting for mymensingh area of bangladesh," *International Journal of Electrical and Computer Engineering*, vol. 15, no. 3, pp. 99–103, 2021.
[12] M. R. Islam, A. Al Mamun, M. Sohel, M. L. Hossain, and M. M. Uddin, "Lstm-based electrical load forecasting for chattogram city of bangladesh," in *2020 International Conference on Emerging Smart Computing and Informatics (ESCI)*. IEEE, 2020, pp. 188–192.
[13] S. H. Rafi *et al.*, "Highly efficient short term load forecasting scheme using long short term memory network," in *2020 8th International Electrical Engineering Congress (iEECON)*. IEEE, 2020, pp. 1–4.
[14] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 6, no. 02, pp. 107–116, 1998.
[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[16] C. Olah, "Understanding lstm networks," 2015, accessed: 2021-10-30. [Online]. Available: http://colah.github.io/posts/2015-08-Understanding-LSTMs/
[17] M. Alhussein, K. Aurangzeb, and S. I. Haider, "Hybrid cnn-lstm model for short-term individual household load forecasting," *IEEE Access*, vol. 8, pp. 180 544–180 557, 2020.
[18] C. Olah, "Conv nets: A modular perspective," 2014, accessed: 2021-10-30. [Online]. Available: https://colah.github.io/posts/2014-07-Conv-Nets-Modular/
[19] "Power grid company of bangladesh ltd." http://pgcb.gov.bd/, accessed: 2021-10-30.