

Electrical and Computer Engineering

Deep Learning Techniques in Load Forecasting

A proposal in partial fulfillment of the MScE

|  |  |  |
| --- | --- | --- |
|  |  | Version <4.1>  Created: 2020 Dec-14  Updated: 2021-Aug-02 |

|  |  |  |
| --- | --- | --- |
|  | Supervised By: | Tolulope Olugbenga  Dr. Dawn MacIsaac, PhD  Dr. Julian Cardenas, PhD |

Table of Contents

[1 Load Forecasting Overview 2](#_Toc73969378)

[2 Investigation 2](#_Toc73969379)

[2.1 The Benchmark Algorithms 2](#_Toc73969380)

[2.1-a Seasonal Naïve Forecaster 2](#_Toc73969381)

[2.1-b Multiple Linear Regression Forecaster 2](#_Toc73969382)

[2.1-c Auto-Regressive Integrated Moving Average with Exogenous Variables 2](#_Toc73969383)

[2.1-d Artificial Neural Network Short Term Load Forecaster – Generation Three 2](#_Toc73969384)

[2.2 Deep Learning Algorithms 2](#_Toc73969385)

[2.3 Metrics for Evaluation 2](#_Toc73969386)

[3 Contributions 2](#_Toc73969387)

[4 References 2](#_Toc73969388)

Table of Figures

[Figure 1:- The Block Diagram of the third generation ANNSTLF [25] 2](#_Toc70354493)

Deep Learning Techniques in Load Forecasting

Updated: 2021-Aug-02 by Tolulope Olugbenga

# Load Forecasting Overview

Load forecasting has been used to plan and operate electric grids for over a century. To ensure a steady supply of electricity, reserve power must be stored. Load aggregators, power marketers, independent system operators, regulatory commissions, industrial/commercial companies, banks, trading firms, and insurance companies also benefit from load forecasting [1]–[5].   Creating a forecasting model for a specific power network is not trivial [4]–[6]. Load forecasts can be affected by weather, time of day, week, and other variables (i.e., coronavirus outbreak). Forecasting electricity demand can be made in several ways: very short-term (VSTLF) (1 day), short-term (STLF) (2 weeks), medium-term (MTLF) (3 years), and long-term (LTLF > 3 years) [7]. Recent research has focused on STLF [1], [7], [8].

Both statistical and machine learning (ML) techniques have been used to forecast load, and the distinction between the two is blurring [1]. Statistical techniques to forecast electrical load include ARIMA modelling [9], [10], and multiple linear regression analysis [11], [12]. However, ML algorithms are more intelligent and can adapt to non-linear and complex relationships between load and other influencing factors (weather, time of day) [7]. A few examples are Artificial Neural Networks (ANNs) [13], [14], Fuzzy Regression Models (FRM) [15], [16], SVMs (support vector machines) [17]. Deep learning approaches like recurrent neural networks (RNN) [18], long-short-term memory networks (LSTM) [19], and 1-D convolution neural networks (CNN) [3], [20] are appealing to researchers in this field because they can learn about temporal dependencies in data inputs. Tao Hong talked about the best technique myth [1]. He concluded that no one technique is best. This depends on the dataset and forecasting needs. No single method will likely work in all load forecasting scenarios. Forecast errors vary greatly between utilities, zones, and horizons. This study compares deep learning forecasting to some conventional forecasters used by utilities to see if deep learning can better meet their needs.

# Investigation

An analysis of deep learning forecasting accuracy compared to current utility forecasting accuracy is presented. The STLF horizons will be the focus. Three data sets will be analyzed. The data from an Independent Electrical System Operator in Ontario were included to aid reproducibility. From 2010 to 2019, both sets cover ten years of hourly city-wide load aggregation measurements from Ottawa and Toronto [21]. The third dataset from St. John Energy is part of a larger Smart Grid Technologies project at UNB. This dataset matches the hourly city-wide load aggregates for 3.5 years (2018 to now). In this work, we will use temperature data from Environment Canada [22]. Four benchmark forecasters will be compared: seasonal naive, MLR, ARIMA, and shallow ANN. Researchers and utilities have used these benchmark algorithms for years [1], [4]–[6], [23]–[25].

The project has three stages. First, we will build the benchmark algorithms. Then a CNN or another deep learning algorithm is used. Finally, deep learning forecasters’ performance will be compared to benchmark algorithms’ performance using available data sets. We will compare overall and peak detection accuracy. The phases are further described below. See the Gantt chart in the appendix for an overview of completed and pending tasks.

## The Benchmark Algorithms

Many publications lack detailed experimental setup information, making direct comparisons with reported results difficult. The benchmark algorithms chosen for this work are relevant and reproducible [1], [4]–[6], [23]–[25].

### Seasonal Naïve Forecaster

The naive forecaster is a widely used benchmark for developing more sophisticated forecasters [23], [26]–[29]. When a naive forecaster outperforms a complex model, we know the complex model is not viable. Bracale [30] et al. state that “the simplest method to anticipate the next value in a time series is to assume it will have the same values as the current value.” However, by taking seasonal trends into account, the Seasonal Naive Forecaster (SNF) improves the naïve forecaster [32]. The SNF is mathematically expressed as (1):

where  are the time series and m is the seasonal period (m=24 for hourly data if taken the day before). This formula uses the previous season’s value. This makes it ideal for predicting variables that are generally stable or vary consistently. It is ineffective at forecasting time series data that fluctuate or are subject to irregularities such as temperature [27].

### Multiple Linear Regression Forecaster

MLR is a statistical technique that is commonly used in load forecasting [8], [12], [15], [23], [30]–[35]. MLR forecasters model continuous dependent variables with multiple independent variables. An MLR with two independent variables is shown below:

In load forecasting, is the load, and  are independent variables like temperature and time of day, s are coefficients estimated, and is an error term. The error term has a constant variance and a mean of 0 [12]. The relationships between the data and the independent variables determine MLR accuracy. Increasing the number of relevant independent variables improves predictive accuracy, but only marginally. MLRs can also simulate non-linear relationships, but only with direct user input [42]. Also, MLRs are incapable of intelligently learning and adapting to newer factors.

### Auto-Regressive Integrated Moving Average (ARIMA)

Any variable that contains data from previous time steps is called a lag feature. Like linear regression, auto-regression uses past values (lag values) as predictors. ARIMA does this by combining AR with a moving average (MA) model [36]. The result is an estimate based on a linear combination of weighted lagged values and lagged errors as delineated in (3) [37]–[39]:

Here  is estimated to account for the average change between consecutive observations, the lag operator is the nth lag value of the time series, is the nth lag error of the time series. Typically, the error terms are supposed to be independently distributed, uniformly distributed variables with a mean of zero. and are the parameters of the autoregressive and moving average parts, respectively. The AR order, p, the MA order, q, and the differencing order, d, are all model parameters. Because linear regression models work best with stationary signals [35], [40], differencing is required. Fernandez et al. used ARIMA, polynomial, neural network, and SVM models to forecast energy load for non-residential buildings [41]. The study used data from the University of Deusto in Donostia-San Sebastian, Spain. The goal was to forecast six days ahead of time. Among the other models, the ARIMA model had the lowest MAPE. They noted that the ARIMA model runs 200 times faster than the SVM model due to fewer parameters.

### Artificial Neural Network Short Term Load Forecaster – Generation Three

The ANNSTLF [1], [24], [35] is a popular ML load forecaster. We will use the third-generation design (G3) [42], which uses two shallow multi-layer feed-forward ANNs with a recursive least squares (RLS) combiner to predict short-term load. The system block diagram is shown below:

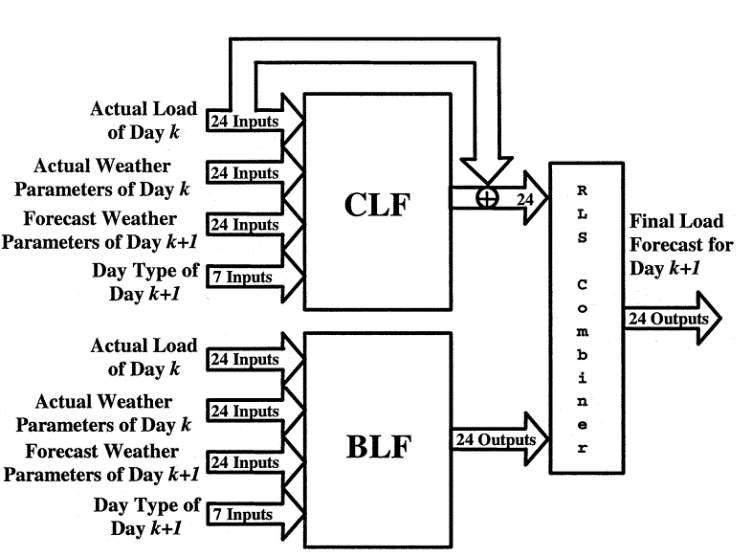


Figure 1:- The Block Diagram of the third generation ANNSTLF [42]

Two multi-layer perceptrons are trained by error back-propagation. The base-load forecaster (BLF) forecasts regular next-day load, while the change-load forecaster (CLF) forecasts daily changes in load demand. The CLF forecaster allows the model to quickly adapt to temperature changes [35], [43], [44]. Both blocks (Figure 1) output a 24x1 vector representing hourly forecasts. To calculate the CLF’s output, it adds predicted changes to last-day values. A weighted average of each block’s output is calculated using an RLS algorithm in the final forecast. In the same utility, a neural network-based approach [45] was developed alongside a regression-based approach [46]. Both models were validated using peak and hourly loads from 1986 to 1990. The ANN model improved peak load and hourly forecasting accuracy. The ANNSTLF-G3 has improved prediction accuracy and generated economic benefits for over a dozen utilities [47], [48]. In some publications [1], [35], ANNSTLF-G3 is the best short-term forecaster.

## Deep Learning Algorithms

The RNN added memory to neural networks, allowing them to model sequential data. However, RNNs are vulnerable to vanishing or exploding gradients [20], [49]. This flaw led to the creation of the LSTM network. The LSTM provides a model that can store information longer and control gradients better. Its memory cell configuration makes it superior to other deep neural networks, said Munem [50] et al. Our UNB graduate student used the LSTM algorithm for a similar project with load forecasting. We will modify his current implementation to fit our datasets and input features.

In load forecasting, convolutional neural networks (CNNs) have also gained popularity [3], [51]–[54]. The CNN is a feed-forward network designed to process data in a grid topology [3], [55]. However, 1D CNNs can be used on time-series data [3], [56]–[58]. CNNs have deeper layers and model parameters like receptive field length and dilation, which can help interpret load data better [20], [59]. Amaradinghe[3] et al. compared the CNN to LSTM, SVM, ANN, and other algorithms for individual building load forecasting. They concluded that CNN is a viable method for predicting load. To create the CNN, we created a Base Load Forecaster, a Change in the Load Forecaster, and an RLS combiner to mimic the ANNSTLF structure [1], [35]. The inputs and structure are the same as the ANNSTLF, but the BLF and CLF components are trained using CNNs.

## Metrics for Evaluation

Mean Absolute Percent Error (MAPE), Mean Bias Error (MBE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Standard Deviation (SD) are all commonly used performance metrics for load forecasting.  The MAPE metric is widely used in load forecasting [1]. When the actuals are zero, MAPE returns undefined values. The algorithm penalizes negative errors more severely than positive errors near zero; this is because low forecasts have a maximum percentage error of 100%, whereas high forecasts have no such limit. MBE measures the model’s overall bias and whether it over-or under-estimates (MBE > or < 0). A forecast model can be highly accurate while remaining biased due to the offset effect of positive and negative error pairs. A significantly biased forecast indicates a model flaw.

MAE is the average magnitude of forecast errors. An absolute fit is measured by the RMSE of the observed and expected values. With the MAE, the error amount is not always noticeable, and the difference between major and minor errors can be hard to tell. The mean absolute error as a percentage was included to address this (MAPE). The MAE and MAPE may underestimate rare but significant errors. We risk missing a massive error by focusing solely on the mean. In order to account for severe errors, we included RMSE. By squaring the errors before computing their mean and then taking the square root of the mean, we get an error size measure that favours significant but rare errors above the mean. Finally, standard deviation measures the spread of errors by comparing them to the mean. The standard deviation is a good measure of dispersion. Time series irregularities have less impact on standard deviation. Extreme values in the time series strongly influence the standard deviation. Unlike other dispersion measures, the standard deviation is difficult to compute and understand. All these are simple tools for assessing forecast accuracy, but they have limitations.

This study will compare performance metrics for each forecaster globally, across forecasts and subsets such as weekdays and weekends, mornings and evenings. It will show us when forecasters perform better or worse than expected.

# Contributions

This research will allow researchers to compare the value added by deep learning algorithms (like CNN and LTSM) to more traditional algorithms. We want to create an algorithm (or a set of algorithms) that can easily adjust to annual increases in power demand, temperature shifts, and other random variables. We also want to develop algorithms that can understand and interpret complex data relationships without explicit user input. This project will also be a reproducible experiment for future researchers. Two of our datasets come from an independent system operator, and the benchmark algorithms we will use are well-documented.

# References

[1] T. Hong and S. Fan, “Probabilistic electric load forecasting: A tutorial review,” *Int. J. Forecast.*, vol. 32, no. 3, pp. 914–938, 2016, doi: 10.1016/j.ijforecast.2015.11.011.

[2] S. Singh, S. Hussain, and M. A. Bazaz, “Short term load forecasting using artificial neural network,” 2018, doi: 10.1109/ICIIP.2017.8313703.

[3] K. Amarasinghe, D. L. Marino, and M. Manic, “Deep neural networks for energy load forecasting,” 2017, doi: 10.1109/ISIE.2017.8001465.

[4] J. Zhang, Y. M. Wei, D. Li, Z. Tan, and J. Zhou, “Short term electricity load forecasting using a hybrid model,” *Energy*, 2018, doi: 10.1016/j.energy.2018.06.012.

[5] C. Kuster, Y. Rezgui, and M. Mourshed, “Electrical load forecasting models: A critical systematic review,” *Sustainable Cities and Society*. 2017, doi: 10.1016/j.scs.2017.08.009.

[6] I. K. Nti, M. Teimeh, O. Nyarko-Boateng, and A. F. Adekoya, “Electricity load forecasting: a systematic review,” *J. Electr. Syst. Inf. Technol.*, 2020, doi: 10.1186/s43067-020-00021-8.

[7] Z. Deng, B. Wang, Y. Xu, T. Xu, C. Liu, and Z. Zhu, “Multi-scale convolutional neural network with time-cognition for multi-step short-Term load forecasting,” *IEEE Access*, vol. 7, pp. 88058–88071, 2019, doi: 10.1109/ACCESS.2019.2926137.

[8] T. Hong, J. Wilson, and J. Xie, “Long term probabilistic load forecasting and normalization with hourly information,” *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 456–462, 2014, doi: 10.1109/TSG.2013.2274373.

[9] L. Tang, Y. Yi, and Y. Peng, “An ensemble deep learning model for short-term load forecasting based on ARIMA and LSTM,” 2019, doi: 10.1109/SmartGridComm.2019.8909756.

[10] B. Nepal, M. Yamaha, A. Yokoe, and T. Yamaji, “Electricity load forecasting using clustering and ARIMA model for energy management in buildings,” *Japan Archit. Rev.*, 2020, doi: 10.1002/2475-8876.12135.

[11] S. Kumar, S. Mishra, and S. Gupta, “Short term load forecasting using ANN and multiple linear regression,” 2016, doi: 10.1109/CICT.2016.44.

[12] A. Y. Saber and A. K. M. R. Alam, “Short term load forecasting using multiple linear regression for big data,” *2017 IEEE Symp. Ser. Comput. Intell. SSCI 2017 - Proc.*, vol. 2018-Janua, pp. 1–6, 2018, doi: 10.1109/SSCI.2017.8285261.

[13] A. Badri, Z. Ameli, and A. Motie Birjandi, “Application of artificial neural networks and fuzzy logic methods for short term load forecasting,” 2012, doi: 10.1016/j.egypro.2011.12.965.

[14] P. H. Kuo and C. J. Huang, “A high precision artificial neural networks model for short-Term energy load forecasting,” *Energies*, 2018, doi: 10.3390/en11010213.

[15] T. Hong and P. Wang, “Fuzzy interaction regression for short term load forecasting,” *Fuzzy Optim. Decis. Mak.*, 2014, doi: 10.1007/s10700-013-9166-9.

[16] M. Hanmandlu and B. K. Chauhan, “Load forecasting using hybrid models,” *IEEE Trans. Power Syst.*, 2011, doi: 10.1109/TPWRS.2010.2048585.

[17] A. Yang, W. Li, and X. Yang, “Short-term electricity load forecasting based on feature selection and Least Squares Support Vector Machines,” *Knowledge-Based Syst.*, 2019, doi: 10.1016/j.knosys.2018.08.027.

[18] 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 Trans. Smart Grid*, vol. 10, no. 1, pp. 841–851, 2019, doi: 10.1109/TSG.2017.2753802.

[19] J. Zheng, C. Xu, Z. Zhang, and X. Li, “Electric load forecasting in smart grids using Long-Short-Term-Memory based Recurrent Neural Network,” 2017, doi: 10.1109/CISS.2017.7926112.

[20] A. Almalaq and G. Edwards, “A review of deep learning methods applied on load forecasting,” *Proc. - 16th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2017*, vol. 2017-Decem, pp. 511–516, 2017, doi: 10.1109/ICMLA.2017.0-110.

[21] “IESO - Hourly Zonal Demand Report.” http://reports.ieso.ca/public/DemandZonal/ (accessed Jun. 05, 2021).

[22] “Historical Climate Data - Climate - Environment and Climate Change Canada.” https://climate.weather.gc.ca/ (accessed Jan. 05, 2021).

[23] T. Hong, P. Wang, and H. L. Willis, “A naïve multiple linear regression benchmark for short term load forecasting,” 2011, doi: 10.1109/PES.2011.6038881.

[24] K. Methaprayoon, W. J. Lee, S. Rasmiddatta, J. R. Liao, and R. J. Ross, “Multistage artificial neural network short-term load forecasting engine with front-end weather forecast,” *IEEE Trans. Ind. Appl.*, 2007, doi: 10.1109/TIA.2007.908190.

[25] A. K. Singh, Ibraheem, S. Khatoon, M. Muazzam, and D. K. Chaturvedi, “Load forecasting techniques and methodologies: A review,” 2012, doi: 10.1109/ICPCES.2012.6508132.

[26] G. Papacharalampous, H. Tyralis, and D. Koutsoyiannis, “Predictability of monthly temperature and precipitation using automatic time series forecasting methods,” *Acta Geophys.*, 2018, doi: 10.1007/s11600-018-0120-7.

[27] P. Wang, B. Liu, and T. Hong, “Electric load forecasting with recency effect: A big data approach,” *Int. J. Forecast.*, 2016, doi: 10.1016/j.ijforecast.2015.09.006.

[28] A. Bracale, G. Carpinelli, P. De Falco, and T. Hong, “Short-term industrial load forecasting: A case study in an Italian factory,” 2017, doi: 10.1109/ISGTEurope.2017.8260176.

[29] M. Rana and I. Koprinska, “Forecasting electricity load with advanced wavelet neural networks,” *Neurocomputing*, 2016, doi: 10.1016/j.neucom.2015.12.004.

[30] T. Hong, M. Gui, M. E. Baran, and H. L. Willis, “Modeling and forecasting hourly electric load by multiple linear regression with interactions,” *IEEE PES Gen. Meet. PES 2010*, pp. 1–8, 2010, doi: 10.1109/PES.2010.5589959.

[31] M. Abuella and B. Chowdhury, “Solar power probabilistic forecasting by using multiple linear regression analysis,” 2015, doi: 10.1109/SECON.2015.7132869.

[32] K. Panklib, C. Prakasvudhisarn, and D. Khummongkol, “Electricity Consumption Forecasting in Thailand Using an Artificial Neural Network and Multiple Linear Regression,” *Energy Sources, Part B Econ. Plan. Policy*, 2015, doi: 10.1080/15567249.2011.559520.

[33] X. Sun, Z. Ouyang, and D. Yue, “Short-term load forecasting based on multivariate linear regression,” 2017, doi: 10.1109/EI2.2017.8245401.

[34] G. Dudek, “Pattern-based local linear regression models for short-term load forecasting,” *Electr. Power Syst. Res.*, 2016, doi: 10.1016/j.epsr.2015.09.001.

[35] R. Weron, *Modeling and forecasting electricity loads and prices: A statistical approach*. wiley, 2006.

[36] K. Goswami, A. Ganguly, and A. K. Sil, “Day ahead forecasting and peak load management using multivariate auto regression technique,” *Proc. 2018 IEEE Appl. Signal Process. Conf. ASPCON 2018*, no. 1, pp. 279–282, 2018, doi: 10.1109/ASPCON.2018.8748661.

[37] G. N. Shilpa and G. S. Sheshadri, “ARIMAX Model for Short-Term Electrical Load Forecasting,” *Int. J. Recent Technol. Eng.*, 2019, doi: 10.35940/ijrte.d7950.118419.

[38] H. Cui and X. Peng, “Short-Term City Electric Load Forecasting with Considering Temperature Effects: An Improved ARIMAX Model,” *Math. Probl. Eng.*, 2015, doi: 10.1155/2015/589374.

[39] A. Shadkam, “USING SARIMAX TO FORECAST ELECTRICITY DEMAND AND CONSUMPTION,” 2020.

[40] R. Bonetto and M. Rossi, “Parallel multi-step ahead power demand forecasting through NAR neural networks,” *2016 IEEE Int. Conf. Smart Grid Commun. SmartGridComm 2016*, pp. 314–319, Dec. 2016, doi: 10.1109/SmartGridComm.2016.7778780.

[41] I. Fernández, C. E. Borges, and Y. K. Penya, “Efficient building load forecasting,” 2011, doi: 10.1109/ETFA.2011.6059103.

[42] A. Khotanzad, R. Afkhami-Rohani, and R. Af, “ANNSTLF - Artificial neural network short-term load forecaster - generation three,” *IEEE Trans. Power Syst.*, vol. 13, no. 4, pp. 1413–1422, 1998, doi: 10.1109/59.736285.

[43] A. Khotanzad, E. Zhou, and H. Elragal, “A neuro-fuzzy approach to short-term load forecasting in a price-sensitive environment,” *IEEE Trans. Power Syst.*, vol. 17, no. 4, pp. 1273–1282, Nov. 2002, doi: 10.1109/TPWRS.2002.804999.

[44] P. R. J. Campbell and K. Adamson, “Methodologies for load forecasting,” 2006, doi: 10.1109/IS.2006.348523.

[45] A. D. Papalexopoulos, S. Hao, and T. M. Peng, “An implementation of a neural network based load forecasting model for the EMS,” *IEEE Trans. Power Syst.*, 1994, doi: 10.1109/59.331456.

[46] A. D. Papalexopoulos and T. C. Hesterberg, “A regression-based approach to short-term system load forecasting,” *IEEE Trans. Power Syst.*, 1990, doi: 10.1109/59.99410.

[47] T. Hong, “Short Term Electric Load Forecasting,” North Carolina State University, 2010.

[48] B. F. Hobbs, “Analysis of the value for unit commitment of improved load forecasts,” *IEEE Trans. Power Syst.*, 1999, doi: 10.1109/59.801894.

[49] D. L. Marino, K. Amarasinghe, and M. Manic, “Building energy load forecasting using Deep Neural Networks,” *IECON Proc. (Industrial Electron. Conf.*, pp. 7046–7051, 2016, doi: 10.1109/IECON.2016.7793413.

[50] M. Munem, T. M. Rubaith Bashar, M. H. Roni, M. Shahriar, T. B. Shawkat, and H. Rahaman, “Electric power load forecasting based on multivariate LSTM neural network using bayesian optimization,” *2020 IEEE Electr. Power Energy Conf. EPEC 2020*, vol. 3, 2020, doi: 10.1109/EPEC48502.2020.9320123.

[51] H. J. Sadaei, P. C. de Lima e Silva, F. G. Guimarães, and M. H. Lee, “Short-term load forecasting by using a combined method of convolutional neural networks and fuzzy time series,” *Energy*, 2019, doi: 10.1016/j.energy.2019.03.081.

[52] I. Koprinska, D. Wu, and Z. Wang, “Convolutional Neural Networks for Energy Time Series Forecasting,” 2018, doi: 10.1109/IJCNN.2018.8489399.

[53] M. Vos, C. Bender-Saebelkampf, and S. Albayrak, “Residential Short-Term Load Forecasting Using Convolutional Neural Networks,” 2018, doi: 10.1109/SmartGridComm.2018.8587494.

[54] W. He, “Load Forecasting via Deep Neural Networks,” 2017, doi: 10.1016/j.procs.2017.11.374.

[55] G. H. Yann LeCun, Yoshua Bengio, “Deep learning (2015), Y. LeCun, Y. Bengio and G. Hinton,” *Nature*, 2015.

[56] R. Fukuoka, H. Suzuki, T. Kitajima, A. Kuwahara, and T. Yasuno, “Wind Speed Prediction Model Using LSTM and 1D-CNN,” *J. Signal Process.*, 2018, doi: 10.2299/jsp.22.207.

[57] A. Brunel *et al.*, “A CNN adapted to time series for the classification of Supernovae,” 2019, doi: 10.2352/ISSN.2470-1173.2019.14.COLOR-090.

[58] N. Singh, C. Vyjayanthi, and C. Modi, “Multi-step Short-term Electric Load Forecasting using 2D Convolutional Neural Networks,” 2020, doi: 10.1109/HYDCON48903.2020.9242917.

[59] A. Gasparin, S. Lukovic, and C. Alippi, “Deep Learning for Time Series Forecasting: The Electric Load Case,” 2019, [Online]. Available: http://arxiv.org/abs/1907.09207.

# Appendix

## Gantt Chart

