Energy Consumption Forecasting for Residential Buildings

***Abstract***–**Electricity consumption is the main form of energy used in buildings. Since the vast majority of electricity is used at the same time it is generated, it is important for utility companies to know the volume of energy needed throughout the day, week, or year. Furthermore, through smart meters, building owners can understand the energy trends of their homes which can help them save money. The implementation of smart meters has allowed buildings to accurately monitor many parameters influencing energy consumption. In this paper, deep learning algorithms of feedforward neural networks, long short-term memory recurrent neural networks, and convolutional neural networks are proposed to forecast energy consumption for the next hour by taking both historical energy consumption data and microclimate data of a residential building. Additionally, feature engineering is performed to improve the accuracy of the model. The findings of this study demonstrate that the feedforward neural network outperforms both the long-short term recurrent neural network and the convolutional neural network.**

Matthew Vandewiel, Jianping Ye, Lichuan Zhang

*Department of Electrical and Computer Engineering*

Western University, London, Ontario, Canada, N6A 5B9

**Key words: Energy consumption forecasting, feedforward neural network, recurrent neural network, convolutional neural network, machine learning, smart meters**

# I. INTRODUCTION

Rapid population growth is fueling the need for more buildings, leading to accelerating levels of global energy consumption [1]. The residential sector accounts for a large portion of this, consuming 27% of the world’s energy [2]. When electricity is generated, almost all of it is used immediately [3], meaning that companies which provide the energy need to know how much electricity to produce at certain times so there is neither an excess nor a shortage. Electricity can be stored, but currently it is inconvenient and expensive compared to using the electricity as it is generated. Therefore, energy consumption forecasting is an appropriate solution which can help utility companies predict the volume and trend of future energy use, allowing them to optimize their operations.

There are two main approaches to energy forecasting: the physical modeling approach and the data-driven approach [4]. However, the data-driven approach is more convenient for consumers and, in addition, provides benefits for them as well as the environment. One application of this approach involves smart meters. Smart meters have been a large part of the Green Button initiative, which allows users to see their energy usage and apply appropriate changes to increase efficiency [5]. The main goal of this project is to help consumers save money, but there is a strong secondary benefit as well: decreasing greenhouse gases. The data from the smart meters, along with weather stations and other sources, are combined to create an algorithm that can forecast energy consumption. This technique is called a sensor-based forecasting approach and is becoming more and more common due to its easily obtained attributes compared to the physical modeling approach [4, 6]. Although more data is better, utility companies do not necessarily need to monitor all houses in an area. By assessing the energy consumption patterns of one residential household, these companies can potentially provide personalized service for an entire residential area [7].

The energy consumption data for this project is from an anonymous residential house in London, Ontario, Canada which is monitored by smart meters. Additionally, weather data is used with the smart meter data to help predict energy consumption.

The goal of this paper was to forecast energy consumption of residential houses. This is achieved through the use of three deep learning algorithms: feedforward neural network (FFNN), long short-term memory recurrent neural network (LSTM-RNN), and convolutional neural network (CNN). As mentioned above, the attributes for these networks include historical energy consumption and weather data.

The remaining sections of this paper is organized as follows: Section II provides an overview on algorithms and performance measures used; Section III describes the related work of other studies; Section IV explains the data and machine learning methodology; Section V presents the results and discussion of the deep learning algorithms; Section VI provides conclusions.

# II. Background

## Feedforward Neural Network

One of the algorithms that was used for energy consumption forecasting was a deep FFNN. Neural networks are an artificial network which attempt to imitate the workings of the human brain. These networks have layers of interconnected neurons that are used to approximate a function [8]. The layers include an input layer which contains all of the attributes, followed by hidden layers which transform the inputs into a form that the third type of layer, the output layer, can understand to provide accurate output(s). Each of the neurons in the layers receives a combination of weights and applies an activation function to return an output. The name “feedforward” is used because the data flows in only one direction: forward from the input layer to the output layer. Also, a “deep” neural network means that there is more than one hidden layer.

## Long Short-Term Memory Recurrent Neural Network

Given that the data is time dependent, using a recurrent neural network (RNN) is logical. Like FFNNs, RNNs are a type of neural network, meaning they also have interconnected neurons through three types of layers: input, hidden, and output. However, the main difference is that RNNs include a feedback loop which takes the output of one or more previous time-steps, and uses these as an input for the output of the next time-step. Furthermore, other attributes can be added, such as weather conditions. Therefore, an output would be dependent on both the past time-step(s) and added attributes. Fig. 1 provides a diagram of an unrolled RNN.

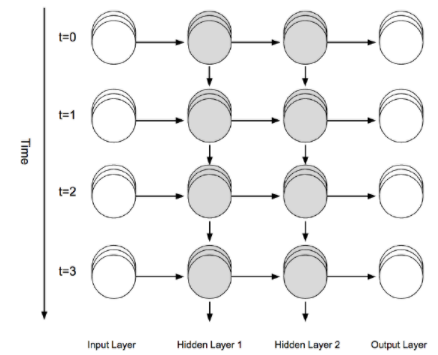


Fig. 1: Recurrent Neural Network Unrolled in the Time Axis [10]

RNNs begin to perform poorly when the gap between the current time-step and the required past time-steps become too large [9]. This is where long short-term memory (LSTM) networks become a useful addition. These networks are able to “remember” information for a long period of time, allowing RNNs to be able to use a large amount of previous information. LSTMs have three units: a forget gate, which decides what information should be kept or discarded; an input gate, which protects the unit from irrelevant inputs; an output gate, which exposes the contents of the memory cell at the output of the LSTM unit. The gating structure allows information from many past time steps to be obtained and also deals with the vanishing gradient problem [10].

## Convolutional Neural Network

CNNs are generally known for their image processing capabilities as a result of their ability to capture spatial and temporal dependencies through relevant filters. This network contains two main types of layers: a convolutional layer and a pooling layer. Convolutional layers use filters to transform the input data into a specific kind of information, such as the edge of an image [10]. Pooling layers reduce the data spatially by using max pooling to extract the largest number within a filter which helps to control overfitting.

## Performance Measures

For both the performance measures applied, the variables are the same: is the number of observations, is the actual energy consumption, and is the predicted energy consumption.

*1) Mean Absolute Percentage Error:* Mean absolute percentage error (MAPE) is a percentage error where the average of the absolute differences between predicted and actual values divided by the actual value is taken, and multiplied by 100, resulting in a percentage of error. The formula for the MAPE is as follows:

(1)

*2) Root Mean Square Error:* Root mean square error (RMSE) is a scale-dependent error that involves rooting the average of the squared difference between predicted and actual values, resulting in a numerical value. The formula for the RMSE is as follows:

(2)

# Related Works

There have been many studies relating to energy consumption and how to predict it in the future. Two main approaches exist: physical modeling and data-driven modeling [4]. In each of these approaches, studies look at different techniques to adjust both data and algorithms for the goal of providing an accurate model.

Physical modeling, also known as engineering methods, relies on physical principles for detailed energy modeling and analysis [4, 11]. Building energy consumption is predicted from variables such as the building and outdoor environment, heating, ventilation, and air conditioning (HVAC) systems, operation schedules, etc. [12]. Physical modeling has its advantages over data-driven techniques: it is specifically based on the science of building physics, there is no data needed, and it’s easy to generalize [13]. One study uses a modified degree-day method to analyze the energy consumption of small buildings focusing on the envelope-based energy and achieves reasonable predictions [14].

Data-driven modeling, as it sounds, is reliant on historical data that can be used within the model. Advantages of this approach include it is fast in computation with real-time data, appropriate for non-linear modeling, and, the biggest advantage, it is often more accurate than physical modeling due to the complex interactions between building systems [13]. Even though an advantage is non-linear modeling, linear and quadratic modeling can be used as well. One study [15] developed both a linear regression model as well as a quadratic regression model from hourly and daily data of a residential building. The external parameters used were outdoor temperature and solar radiation. The study found that using both parameters was important, as well as that the quadratic regression model had better predictions for short periods of time (hours), but not necessarily longer periods (days). Another study by Zhang et al. [7] used support vector regression to forecast energy consumption. This study used weather attributes and past hourly energy consumption data gathered from smart meters. When weather conditions data (“clear”, “cloudy”, etc.) was missing information for an hour, it was replaced with the last recorded hour’s weather condition. For the rest of the weather attributes, such as temperature or humidity, the average value of both the previous and following hours were taken.

Artificial neural networks (ANN) are an effective approach with energy consumption forecasting due to their ability to solve non-linear problems. As a result, these networks are widely used for energy consumption forecasting [16]. Like [7], a different study used weather attributes in combination with historical energy data [17]. However, in this study a FFNN was used. The results were very accurate, as the MAPE was 7.32% for forecasting energy a month in advance. Karatasou et al. [18] also used a FFNN for data consisting of weather and past hourly values of energy consumption. However, the research was performed in two different ways: one with a single-step, which depended on the past data, and another with multi-step, which was independent. Results indicated that the single-step approach had a very accurate prediction, with a lower MAPE than its multi-step counterpart.

RNNs are commonly used due to their ability to model the time dependent historical data. Kong et al. [19] use a LSTM-RNN and obtain their data from smart meters in 69 Australian residential buildings. They focus on short-term load forecasting for each household and find that aggregating the forecasts from all homes yields better results than the traditional direct forecasting of the aggregated load. Similarly, another study [20] in Japan uses historical energy consumption data and historical temperatures from the previous 20 years. It achieves reasonable results for energy forecasting one year in advance.

A new approach combining CNN and LSTM was used in a recent study [21]. This model uses past electricity consumption as an input to forecast future electricity consumption. The accuracy is quite high due to the model being able to extract both spatial and temporal features effectively.

In the current paper, a data-driven approach was taken as it is generally more accurate than a physical approach. Additionally, ANNs are able to solve non-linear problems making them quite effective for energy consumption forecasting. Therefore, deep learning models involving FFNNs, LSTM-RNNs, and CNNs were used. In an attempt to further increase accuracy, this study provides different approaches from those above. CNNs are a relatively new concept for energy consumption forecasting, as few studies use it for this purpose. Moreover, weather appears to be an important factor in determining energy consumption in many of the studies; however, usually very few weather features besides temperature and humidity are used [22]. Therefore, many different weather features will be implemented for the current study. Furthermore, for all models the unit circle technique was used to ensure that the amount of time is consistent from hour to hour, day to day, etc.

# Methodology

This section elaborates on the proposed methods and detailed procedures followed by an analysis and approach to the problem. It is divided into the following sections: data set details, data preprocessing, exploratory data analysis, data preparation, modeling, hyperparameter tuning, and validation process.

## Data Set Details

London Hydro provided the 17,544 meter readings with the day and hour in which it was recorded, as well as the energy consumption in kilowatt-hours (kWh) corresponding to each hour. This data set has two years of values, ranging from July 2018 to July 2020. However, due to COVID-19 forcing many people to begin working from their own home in March 2020 [23], only data up to the end of February 2020 was used, reducing the number of meter readings to 14,616. Weather data is added to the historical energy consumption to help predict future energy consumption. To ensure accuracy, the closest weather station to the house monitored was chosen, and this data was obtained from the official Government of Canada website [24]. The attributes used from the weather station include temperature, dew point temperature, relative humidity, wind direction, wind speed, visibility, and air pressure. These can help to determine the microclimate of the house.

After the initial testing is done, the most accurate algorithm is used for energy consumption forecasting for data in the COVID-19 period, which includes 2,928 meter readings.

## Data Preprocessing

1) Data Cleansing: The merged data set was inspected to clean the missing or invalid entries. Missing data or values with invalid format can severely damage the model to an extent that no constructive output may be computed. The substitution strategy used is forward filling which replaces the missing data with the last available observation.

2) Feature Preparation: Several additional features are engineered and re-constructed from the raw data set as they may contribute to the energy consumption variation.

* Hour of the day: The hour of the day is represented by 1 to 24. The intuition is that some usage peaks might occur at daytime, while at midnight the consumption may be at the trough.
* Day of the week: Monday to Sunday is represented by 1 to 7 in the original data set. It is assumed that weekdays and weekends have different consumption patterns due to life and work style changes.
* Month of the year: Each month is represented by 1 to 12. The month feature is used to magnify signal input indicating high and low temperature periods.
* Usages at previous time steps: Energy consumption readings from previous time steps are added to the current time steps as input variables. These features aim to use lagged observations to facilitate the forecasting of the current time steps.
* Season of the year: The four seasons (spring, summer, fall, and winter) are generated from the month feature. The season factor is believed to have an impact on the usage pattern as it affects the way that electrical appliances are used.
* Temperature: Temperature is an important factor to be considered as it is directly related to the use of air conditioning and heating.
* Dew point temperature: Dew point is related to relative humidity; it gives the minimum temperature needed for vapor to condense.
* Relative Humidity: Relative humidity is the percentage of water vapor that exists compared to maximum amount of water vapor. A house generally has both humidifier and dehumidifier to control the relative humidity to a comfortable range. Furthermore, humidity generally makes humans feel warmer due to decreased evaporative cooling. Both of these factors can cause a rise in electricity consumption.
* Wind direction: Wind direction may allow for natural ventilation of the house, reducing the use of HVAC systems and electricity usage.
* Wind speed: Wind can facilitate the movement of air, and potentially affect the temperature of the house. Thus, wind speed can have a subtle effect on electricity usage.
* Visibility: Visibility is the distance that can be seen from the weather station. Through an exhaustive review of 116 studies of energy consumption forecasting models by Wei et al. [22], only one study used visibility. Therefore, it is used in an attempt to increase accuracy.
* Pressure: Air Pressure affects relative humidity and potentially the wind speed.

## Exploratory Data Analysis

The relationship between weather features and energy consumption is visualized and investigated. From Fig. 2, consumption from July to September 2018 is generally higher than that of November to December 2018, which may be attributable to the use of air conditioners during the summer. In the winter period, during December 2018 to January 2019, the consumption peaks occur frequently, which may be related to the use of space heating devices and increasing demand for lighting. Fig. 3 shows the consumption pattern of the first week of collected data. From this, it is evident that the consumption peak occurs during the weekend.

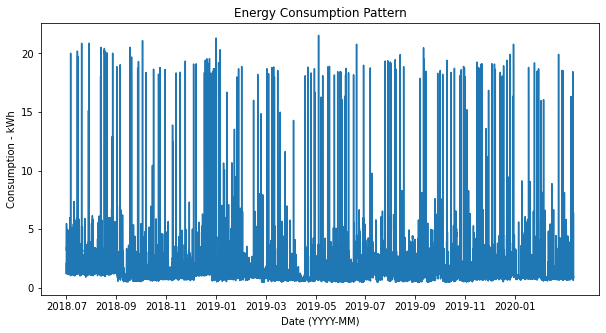


Fig 2: Energy Consumption of House from July 2018 to February 2020

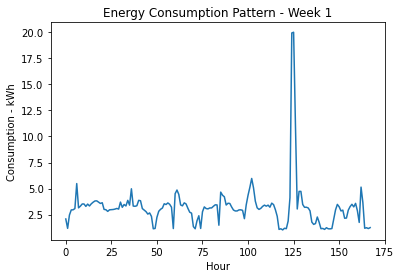


Fig 3: Energy Consumption of House in Week 1

## Data Preparation

### Unit Circle Mapping for Cyclical Features: When working with time-related features, some are usually cyclical. For example, the “hour of the day” feature takes values from 0 to 23. A problem arises when time changes from 23 to 0. The actual time gap is merely one hour apart, but the current encoding interprets the distance as 23. If these cyclical features were to be fed into the machine learning algorithm, data discontinuities may cause unexpected damage. Therefore, an alternative encoding is introduced as follows to ensure equal distancing between consecutive observations of cyclical features. Unit circle mapping utilizes sine and cosine transformation to encode the data into two dimensions and form a unit circle:

(3)

(4)

where x is the cyclical attribute to be transformed. The cyclical features that need to be transformed are the hour of the day, day of the week, month of the year, and season.

2) Data Splitting and Standardization: The transformed data set was divided into training, validation, and test sets with ratios of 60%, 20%, and 20%, respectively. Noticeably, the data is not shuffled so that sequential relationship is preserved. The final training set contains one year of data with 8,769 hourly observations, while the validation and test sets each have four months of data with 2,923 hourly observations.

Standardization is applied so the features in the data set have a mean of zero and unit variance. The mean and standard deviation of each feature are calculated on the training set, which are then applied to scale the validation and test sets to avoid data leakage.

3) Sliding Window: To effectively transform the time-series data set into a supervised learning data set, and to generate additional observations, the data was regrouped into windows that contain samples in multiple time steps. The stride of sliding, which is the number of time steps from one window to the next, was set to one. The number of samples that the window captures was optimized by hyperparameter tuning.

## Modeling

As mentioned earlier, FFNN, LSTM-RNN, and CNN models were developed to compare accuracy. All models have the following components: He initialization, as this is the recommended weight initialization strategy to be used along with ReLU and its variants, such as exponential linear unit (ELU) [25]; ELU activation was selected because it has a solution to the dying ReLU problem and alleviates the vanishing gradients problem during gradient descent; Nadam is the Adam optimizer with Nesterov momentum, and was used as it is computationally efficient due to the accelerated learning process; early stopping was implemented to interrupt training when the validation loss stops decreasing for a number of epochs, and the best model with the lowest validation error was restored at the end of training; finally, mean square loss function was selected as it is a common evaluation method to find the loss in learning curve plots. For the LSTM-RNN and CNN models, a sliding window approach was also used to generate windows of input data and L2 regularization was used at each dense layer to reduce overfitting.

### Feedforward Neural Network: There were 21 neurons in the input layer as there were 21 features to be fed into the network. Next, the number of hidden layers and the associated number of neurons were determined through hyperparameter tuning. Finally, the output layer consisted of a single neuron with linear activation, as one prediction was made for each time step.

### Long Short-Term Memory Recurrent Neural Network: The input for the LSTM-RNN network is a three-dimensional array, with each dimension representing batch size, time steps, and features, where the same features in the FFNN are fed into this network. It contains multiple LSTM layers followed by an output layer with the predicted usage value for the next hour.

### Convolutional Neural Network: This architecture consists of a one-dimensional convolution layer, followed by multiple fully connected layers, and an output layer. The one-dimensional convolution layer slides a certain number of kernels across the input sequence, with each kernel producing a one-dimensional feature map. Each feature map obtained corresponds to a short sequential pattern. The same features as both the FFNN and LSTM-RNN are fed into the network. The number of hidden layers and the number of neurons in these layers were determined by hyperparameter tuning. The output layer was designated as one neuron with linear activation to output the predicted value.

## Hyperparameters Tuning

A randomized search was conducted for each model to optimize performances. A total of 30 combinations were attempted. The hyperparameter combination that yields the lowest validation loss was selected and the corresponding model was used to predict the test set. Hyperparameters implemented for each algorithm are summarized in Table I.

TABLE I

Hyperparameters and Parameter Distributions

|  |  |  |
| --- | --- | --- |
| **Model** | **Hyperparameter** | **Parameter Distributions** |
| FFNN, CNN | Number of Hidden Layers | 3, 4, 5, 6 |
| LSTM-RNN | 1, 2, 3 |
| FFNN, LSTM-RNN, CNN | Number of Neurons Per Layer | 30, 40, 50 |
| FFNN, LSTM-RNN, CNN | Regularization Factor | 0.01, 0.02, 0.03 |
| FFNN, LSTM-RNN, CNN | Batch Size | 16, 32, 64 |
| LSTM-RNN, CNN | Sliding Window Size | 6, 12, 24 |
| CNN | Filter Size | 8, 16, 24 |

## Validation Process

After the model was trained and tuned, it was evaluated on the test set. The predicted values were inverse transformed to the original scale to allow for comparisons. The optimal model was selected based on the minimum MAPE achieved on the test set.

# Results and Discussion

The models were implemented by using the Keras machine learning library with a TensorFlow GPU backend. Table II provides the optimal tuned parameters of each model.

TABLE II

Optimal tuned parameters

|  |  |
| --- | --- |
| **Model** | **Optimal Tuned Parameters** |
| FFNN | Layers = 3; Neurons = 30; Regularization = 0.01; Batch Size = 16 |
| LSTM-RNN | Layers = 3; Neurons = 50; Regularization = 0.01; Batch Size = 32; Window Size = 6 |
| CNN | Layers = 4; Neurons = 50; Regularization = 0.01; Batch Size = 32; Window Size = 6; Filters = 8 |

## Feedforward Neural Network

Fig. 4 shows the predictions for 4 months data starting in November 2019. It is observed that the model is capable of capturing small variations and following the actual consumption pattern. More importantly, the model successfully captures the time steps when higher consumption occurs. However, it lacks sufficiently high accuracy while predicting the peak values, which may be attributable to the regularization penalty applied.

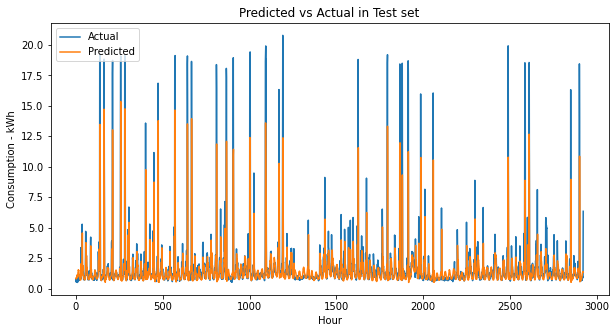


Fig. 4: FFNN Actual vs. Predicted Test Set Results

## Long Short-Term Memory Recurrent Neural Network

Fig. 5 shows the predictions made by the LSTM-RNN in the same set of data. Noticeably, the model leaves many consumption peaks uncaptured. This may due to an unsuitable window size such that the temporal sensitivity is not well captured. The usage pattern can also be intrinsically irregular and act as a source of randomness, making it difficult to predict according to time. Overall, the model follows a weak consumption pattern and lacks accuracy when predicting consumption peaks.

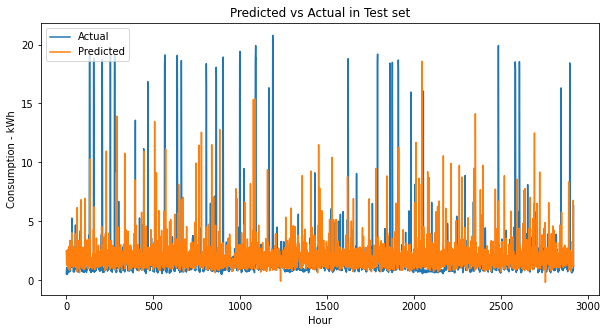


Fig. 5: LSTM-RNN Actual vs. Predicted Test Set Results

## Convolutional Neural Network

In the prediction of the CNN seen in Fig. 6, a similar situation occurs as in the LSTM-RNN. The model is able to predict consumption peaks occasionally, but often comes at the price of low success rates which eventually leads to large penalties from the square loss function. Even worse, the model tends to make large predictions that are in the wrong time steps. This can be inferred from the observation that there is a significant mismatch between predictions and actual data.

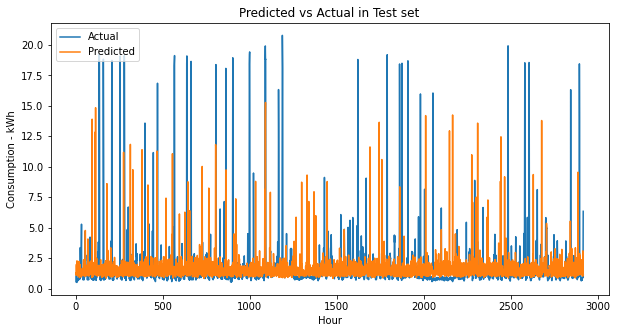


Fig. 6: CNN Actual vs. Predicted Test Set Results

## Comparison of Algorithms

The two comparison plots below (Figs. 7 & 8) show performance metrics of the three models. It is apparent that the FFNN achieves the lowest MAPE and RMSE. This is a result of both the LSTM-RNN and CNN being penalized significantly while predicting consumption peaks, as depicted in the prediction plots (Figs. 5 & 6).

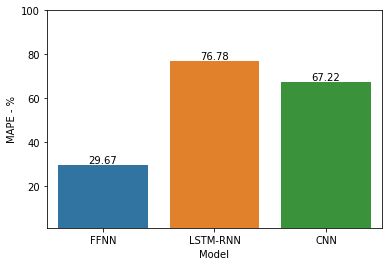


Fig. 7: Comparison Plot of MAPE in Algorithms

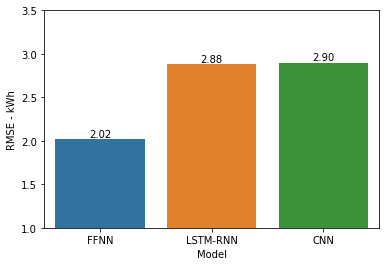


Fig. 8: Comparison Plot of RMSE in Algorithms

## Testing during COVID-19 Period

To examine the generalization ability of the model to a further extent, the second part of the original data set, data from March 2020 to July 2020, was used as an additional test set. To remain consistent between data sets, identical methodologies and procedures were followed to preprocess and transform the data. Specifically, the same feature scaler objects used previously are again used along with the trained and tuned FFNN model. The prediction is shown in Fig. 9.

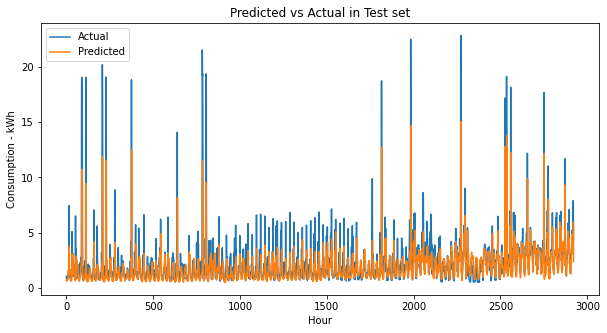


Fig. 9: FFNN Actual vs. Predicted Test Set Results During COVID-19

The MAPE and RMSE achieved are 31.63% and 1.75 kWh, respectively. The results are similar to that of the previous test data before the lifestyle changes due to COVID-19. This additional testing further proves the generalization ability of the FFNN model given the fact that the consumption pattern had indeed changed as shown in Fig 8. For example, there are fewer high consumption readings, and hence less fluctuation. A comparison of the accuracy of the model for both time periods is shown below (Table III).

TABLE III

FFNN Pre-Covid-19 Period vs. Covid-19 Period

|  |  |  |
| --- | --- | --- |
| **Period** | **MAPE (%)** | **RMSE (kWh)** |
| Pre-COVID-19  (Jul. 2018 - Feb. 2020) | 29.67 | 2.02 |
| COVID-19  (Mar. 2020 – Jul. 2020) | 31.63 | 1.75 |

# Conclusion

This study focused on the data-driven approach to forecast residential energy consumption. Specifically, deep learning approaches were taken to construct prediction models. The microclimate data obtained from the official Government of Canada website was merged with energy consumption data to form the final data set. The study started with data cleaning, followed by feature engineering to generate potentially useful features for improving forecasting accuracy. In contrary with other studies, a wider range of weather attributes were used. Also, several time-related features were constructed to amplify sequential ordering signal of the data. Then, exploratory data analysis was conducted to discover and visualize the energy usage pattern. The cyclical features were encoded using unit circle mapping to maintain sequential continuity. Next, the data set was split, standardized, and finally reformatted through windowing. Three deep neural networks were trained, tuned, and then compared. After comparing the models, the most accurate one, FFNN, was then further evaluated on COVID-19 data to test the generalization performance.

From the comparison results, it was unexpected that the FFNN was the best performing model, achieving a MAPE of 29.31% and a RMSE of 2.0 kWh. There are two major potential reasons that could result in the poor performance of LSTM which was anticipated to outperform the FFNN. Firstly, the usage pattern of the investigated household might be irregular such that predicting through time dependency would cause severe biases. Secondly, the selected window size may be too small that temporal behavior was not properly recognized by the model. The irregularities may act as noises to affect the accuracy of LSTM. In contrary, the FFNN does not consider sequential relationships and thus such impacts were avoided.

The result of this work can be further improved. One feasible implementation would be using hybrid model that combines convolution layer with LSTM. Convolution can be used to shorten the input sequences and preserve only higher order information. By doing this, LSTM may be able to learn patterns in much longer sequences, such as the usage pattern for a week.

# References

1. G.Y. Zhao, Z.Y. Liu, Y. He, H.J. Cao, and Y.B. Guo, “Energy consumption in machining: Classification, prediction, and reduction strategy,” *Energy,* vol. 133, pp. 142-157. 2017.
2. P. Nejat, F. Jomehzadeh, M.M. Taheri, M. Gohari, and M.Z.A. Majid, “A global review of energy consumption, CO2 emissions and policy in the residential sector (with an overview of the top ten CO2 emitting countries),” *Renewable and Sustainable Energy Reviews,* vol. 43, pp. 843-862. 2015.
3. H. Ibrahim, A. Ilinca, and J. Perron, “Energy storage systems—Characteristics and comparisons,” *Renewable and Sustainable Energy Reviews,* vol. 12, no. 5, pp. 1221-1250. 2008.
4. K. Amasyali and N.M. El-Gohary, “A review of data-driven building energy consumption prediction studies,” *Renewable and Sustainable Energy Reviews,* vol. 81, pp. 1192-1205. 2018.
5. London Hydro, “My Green Button,” http://mygreenbutton.ca/.
6. R.K. Jain, K.M. Smith, P.J. Culligan, and J.E. Taylor, “Forecasting energy consumption of multi-family residential buildings using support vector regression: Investigating the impact of temporal and spatial monitoring granularity on performance accuracy,”, *Applied Energy,* vol. 123, pp. 168-178. 2014.
7. X.M. Zhang, K. Grolinger, M.A.M. Capretz, and L. Seewald, "Forecasting Residential Energy Consumption: Single Household Perspective," in 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 110-117. 2018.
8. T. Gupta. “Deep Learning: Feedforward Neural Network.” Towards Data Science. https://towardsdatascience.com/deep-learning-feedforward-neural-network-26a6705dbdc7.
9. J. Kumar, R. Goomer, and A.K. Singh, “Long Short Term Memory Recurrent Neural Network (LSTM-RNN) Based Workload Forecasting Model For Cloud Datacenters,” *Procedia Computer Science,* vol. 125, pp. 676-682. 2018.
10. J. Patterson and A. Gibson, “Major Architectures of Deep Networks,” in *Deep learning: a practitioner’s approach,* 1st ed., Beijing, China:O’Reilly Media, 2017, pp. 130-153.
11. H. Zhao and F. Magoulès, “A review on the prediction of building energy consumption,” *Renewable and Sustainable Energy Reviews,* vol. 16, no. 6, pp. 3586-3592. 2012.
12. I. Petri, H. Li, Y. Rezgui, Y. Chunfeng, B. Yuce and B. Jayan, “A modular optimisation model for reducing energy consumption in large scale building facilities,” *Renewable and Sustainable Energy Reviews,* vol. 38, pp. 990-1002. 2014.
13. C. Deb, F. Zhang, J. Yang, S.E. Lee and K.W. Shah, “A review on time series forecasting techniques for building energy consumption,” *Renewable and Sustainable Energy Reviews,* vol. 74, pp. 902-924. 2017.
14. M.S. Al-Homoud, “Computer-aided building energy analysis techniques,” *Building and Environment,* vol. 36, no. 4, pp. 421-433. 2001.
15. N. Fumo and M.A.R. Biswas, “Regression analysis for prediction of residential energy consumption,” *Renewable and Sustainable Energy Reviews,* vol. 47, pp. 332-343. 2015.
16. M.A.M. Daut, M.Y. Hassan, H. Abdullah, H.A. Rahman, M.P. Abdullah and F. Hussin, “Building electrical energy consumption forecasting analysis using conventional and artificial intelligence methods: A review,” *Renewable and Sustainable Energy Reviews,* vol. 70, pp. 1108-1118. 2017.
17. S.M. Islam, S.M. Al-Alawi, and K.A. Ellithy, “Forecasting monthly electric load and energy for a fast growing utility using an artificial neural network,” *Electric Power Systems Research,* vol. 34, no. 1, pp. 1-9. 1995.
18. S. Karatasou, M. Santamouris and V. Geros, “Modeling and predicting building's energy use with artificial neural networks: Methods and results,” *Energy and Buildings,* vol. 38, no. 8, pp. 949-958. 2006.
19. 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, pp.841-851. 2019.
20. B. Kermanshahi, “Recurrent neural network for forecasting next 10 years loads of nine Japanese utilities,” *Neurocomputing,* vol. 23, no. 1-3, pp. 125-133. 1998.
21. T.Y. Kim, S.B. Cho, “Predicting residential energy consumption using CNN-LSTM neural networks,” *Energy,* vol. 182, pp. 72-81. 2019.
22. N. Wei, C. Li, X. Peng, F. Zeng, and X. Lu, “Conventional models and artificial intelligence-based models for energy consumption forecasting: A review,” *Journal of Petroleum Science and Engineering,* vol. 181, pp. 106-187. 2019.
23. S. Davidson. “Working from home in Ontario will be recommended for the ‘foreseeable future’.” CTV News. https://toronto.ctvnews.ca/working-from-home-in-ontario-will-be-recommended-for-the-foreseeable-future-1.5035635.
24. Government of Canada, “Canada weather information,” https://weather.gc.ca/.
25. A. Géron, “Training Deep Neural Networks,” in *Hands-on Machine Learning with Scitkit-Learn, Keras, and Tensorflow,* 2nd ed., Sebastopol, CA, USA: O’Reilly Media, 2019, pp. 327.