# Experiment Setup

## Hardware

The tests were carried out on a PC with a Windows OS. The computer's features are described in the following:

* GPU: Intel Nvidia RTX 2070
* CPU: Intel Core i7 8th Generation
* RAM:
* Mother Board:

## Software

Python is the programming language utilised. The experiment is carried out in a virtual environment generated with Visual Studio Code and Python 3.9.0. The relevant packages are utilised in the experiment. A Jupyter Notebook was used to write and run the code. The network was trained using the Nvidia RTX 2070 GPU, which has 16 GB VRAM and CUDA version 11.0.

* TensorFlow
* Keras
* Scikit-learn
* Pandas
* Prophet
* Matplotlib
* NumPy
* Seaborn

## Settings

# Cases

# Data and Training

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Fold 1 | Train | Test | | | |
| Fold 2 | Train | | Test | | |
| Fold 3 | Train | | | Test | |
| Fold 4 | Train | | | | Test |

# Model Experiments

## Loss Functions

## Activation Functions

# Model Structure

## (How many layers)

# Output of the cases

# Experiments and Results

This chapter will go about performing ‘black-box’ testing to evaluate or created models and afterwards a visualization of the models will be shown ranking by the model with the best performance. All the models will be tested against a baseline prediction calculated using the simple persistence algorithm. The Zero Rule Algorithm is by far the most frequent baseline method for monitoring ML. In the instance of classifications, the algorithm correctly forecasts most classes. With regression cases, it predicts the expected average. This can be utilized with time-series data, but the sequence correlation architecture in the data set is not considered. The persistence algorithm is a similar technique for time series data sets. The persistence method predicts the expected outcome for the next step (t + 1) based on the last step (t-1). The following baseline forecast conditions must be fulfilled:

* **Simple**： A technique that does not need extensive training or understanding.
* **Fast**： Predicting in a quick and computationally simple manner.
* **Repeatable**： A deterministic technique gives the expected output when given the same input.

## Queen’s Building

### Baseline

|  |  |  |  |
| --- | --- | --- | --- |
| Hourly Baseline | | | |
|  | Electric | Gas | Water |
| RMSE | 4.265 | 152110.65 | 0.057 |
| MSE | 18.190 | 23137650036.14 | 0.003 |
| MAE | 2.4481 | 726.80 | 0.034 |
| R2 | 0.944 | -2.000 | 0.895 |

From the table above we can see the gas dataset metrics could not accurately be measured which means a further investigation of the plot is needed.

Chart, line chart

Description automatically generated

The plot of Queen’s Building Gas Hourly Dataset

Based on the figure above we can see that the hourly dataset could not be plotted due to the fact we are dealing with data lower than the program can compute. The electric and water datasets produced very good metrics that could help train the desired algorithm, but with the gas dataset producing such results, it could degrade the performance.

|  |  |  |  |
| --- | --- | --- | --- |
| Daily Baseline | | | |
|  | Electric | Gas | Water |
| RMSE | 8.217 | 53.023 | 0.071 |
| MSE | 67.521 | 2811.472 | 0.005 |
| MAE | 4.996 | 26.286 | 0.043 |
| R2 | 0.644 | 0.540 | 0.744 |

The daily baseline shows very good results and a perfect univariate baseline to try to beat for the chosen algorithms. These are all baseline metrics from a univariate model in hopes that adding regressors would improve it.

### LSTM

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Layers** | **Neurons** | **Epochs** | **Lag Variable** | **Metrics** | **Beaten?** |
| 2 | 50 | 50 | 6 hours | RMSE: 4.089  MSE: 16.723  MAE: 2.639  : 0.944 | Yes |
| 50 | 200 | 6 hours | RMSE: 4.180  MSE: 17.473  MAE: 2.595  : 0.942 | Yes |
| 150 | 50 | 6 hours | RMSE: 5.066  MSE: 25.664  MAE: 3.940  : 0.914 | No |
| 6 | 50 | 50 | 6 hours | RMSE: 9.510  MSE: 90.433  MAE: 7.489  : 0.698 | No |
| 8 | 32 | 50 | 6 hours | RMSE: 11.859  MSE: 140.642  MAE: 9.564  : 0.531 | No |

We can observe from the table above that if the LSTM model contains two layers, it improves the baseline RMSE. Another interesting fact is that the number of neurons is capped at 50; much greater and the RMSE begins to exceed the baseline. Due to time constraints, the remaining tests will be tested using the Daily dataset.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Layers** | **Neurons** | **Epochs** | **Training** | **Metrics** | **Beaten?** |
| 2 | 50 | 50 | 4 years | RMSE: 6.864  MSE: 47.117  MAE: 5.471  : 0.662 | Yes |
| 50 | 500 | 4 years | RMSE: 6.039  MSE: 36.465  MAE: 4.404  : 0.738 | Yes |
| 150 | 500 | 5 years | RMSE: 5.619  MSE: 31.576  MAE: 4.162  : 0.772 | Yes |
| 300 | 1000 | 5 years | RMSE: 6.707  MSE: 44.980  MAE: 4.972  : 0.675 | Yes |
| 1000 | 500 | 5 years | RMSE: 5.546  MSE: 30.761  MAE: 4.247  : 0.778 | Yes |
| 6 | 50 | 50 | 5 years | RMSE: 10.842  MSE: 117.539  MAE: 9.111  : 0.150 | No |
| 50 | 500 | 5 years | RMSE: 5.923  MSE: 35.081  MAE: 4.442  : 0.746 | Yes |
| 150 | 500 | 5 years | RMSE: 7.003  MSE: 49.041  MAE: 5.424  : 0.645 | Yes |
| 300 | 1000 | 5 years | RMSE: 6.623  MSE: 43.861  MAE: 5.032  : 0.683 | Yes |
| 8 | 50 | 500 | 5 years | RMSE: 7.200  MSE: 51.838  MAE: 5.470  : 0.625 | Yes |

Based on the extensive tests carried out, we can see that the LSTM performs very well and improves the RMSE. The LSTM with 2 layers and 1000 input neurons had the better metric. From this, we can assume that the BLSTM would also show similar results or even better.

### BLSTM

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Layers** | **Neurons** | **Epochs** | **Train Variable** | **Metrics** | **Beaten?** |
| 2 | 50 | 50 | 5 years | RMSE: 5.480  MSE: 30.025  MAE: 4.173  : 0.783 | Yes |
| 50 | 200 | 5 years | RMSE: 4.817  MSE: 23.206  MAE: 3.508  : 0.832 | Yes |
| 200 | 200 | 5 years | RMSE: 4.807  MSE: 23.104  MAE: 3.591  : 0.833 | Yes |
| 6 | 50 | 50 | 5 years | RMSE: 7.200  MSE: 51.839  MAE: 5.744  : 0.625 | Yes |
|  | 150 | 500 | 5 years | RMSE: 6.624  MSE: 43.883  MAE: 5.012  : 0.683 | Yes |
| 8 | 50 | 200 | 5 years | RMSE: 11.280  MSE: 127.247  MAE: 9.271  : 0.080 | No |

Based on the extensive tests carried out, we can see that the LSTM performs very well and improves the RMSE. The BLSTM with 2 layers and 200 input neurons had the better metric. This proves the statement made in the literature that for a bidirectional LSTM, only 2 layers are needed for it to work effectively. Therefore, the rest of the test would only be carried out using only 2 layers of both the LSTM and BLSTM.

### XGBoost

### LightGBM

### Prophet

## Hugh Aston

### Baseline

|  |  |  |  |
| --- | --- | --- | --- |
| Hourly Baseline | | | |
|  | Electric | Gas | Water |
| RMSE | 7.713 | 27.48 | 0.131 |
| MSE | 59.483 | 755.156 | 0.017 |
| MAE | 4.646 | 12.024 | 0.083 |
| R2 | 0.917 | 0.827 | 0.853 |

|  |  |  |  |
| --- | --- | --- | --- |
| Daily Baseline | | | |
|  | Electric | Gas | Water |
| RMSE | 13.573 | 25.191 | 0.172 |
| MSE | 184.23 | 634.58 | 0.03 |
| MAE | 8.354 | 14.7 | 0.115 |
| R2 | 0.34 | 0.803 | 0.36 |

### LSTM

### BLSTM

### XGBoost

### LightGBM

### Prophet

## Gateway House

### Baseline

|  |  |  |  |
| --- | --- | --- | --- |
| Hourly Baseline | | | |
|  | Electric | Gas | Water |
| RMSE | 16753.53 | 34.104 | 0.134 |
| MSE | 280680631.65 | 1163.092 | 0.018 |
| MAE | 83.353 | 10.998 | 0.047 |
| R2 | -1.999 | 0.702 | 0.741 |

|  |  |  |  |
| --- | --- | --- | --- |
| Daily Baseline | | | |
|  | Electric | Gas | Water |
| RMSE | 16.602 | 31.317 | 0.124 |
| MSE | 275.639 | 980.762 | 0.015 |
| MAE | 9.609 | 14.89 | 0.072 |
| R2 | 0.54 | 0.411 | 0.271 |

### LSTM

### BLSTM

### XGBoost

### LightGBM

### Prophet