

1. Introduction

Electronic vehicles and a number of other portable electronic devices use Lithium-ion batteries since they are rechargeable. They have a higher energy density than other previously used cadmium or lead batteries.

Also, lithium is lightest of all metals and they are also safer than many alternatives. Battery producers must ensure that consumers are protected in the unlikely case of a battery failure. Maintenance is performed on battery-powered equipment to ensure that it runs smoothly. It is critical to keep track of the status of batteries in order to ensure that these systems work properly. For these systems, multiple maintenance plans are used, such as reactive, preventive and predictive maintenance. In predictive maintenance, data regarding the health of the equipment is collected and accordingly maintenance is scheduled. This method is cost effective, and it decreases the probability of failure.

In the field of batteries, predictive maintenance is used for their health monitoring, which includes prediction of battery's State of Charge (SOC) and Remaining Useful Life (RUL). SOC shows the present capacity of batteries in comparison to its rated capacity. RUL shows how much longer battery is going to last under current working conditions. This project is specifically focused on the prediction of RUL of Li-ion

batteries. Accurate RUL prediction helps in overall maintenance of the system and gives the clear idea about replacement of the batteries.

Previous research projects used basic machine learning models like , Support Vector Machine (SVM), Probabilistic Algorithms, Kalman Filters, Optimization Algorithm etc. to predict the State of Health (SOH), SOC and RUL of the batteries. The deep learning NN–long short-term memory (LSTM) RNN is specifically developed to solve the long-term dependency problem. It is able to remember information for long periods of time via the introduced gates.

At first, the data is cleaned, pre-processed and transformed so that any model can be fitted on to it. A simple RNN model is built as a base model and the performance is evaluated. Later the model is saved and their weights are passed on to different set of data to predict the RUL. And finally, a better model is finalized with comparatively less error.

2. Literature Survey

In the paper “Remaining Useful Life Assessment for Lithium-ion Batteries using CNN-LSTMDNN Hybrid Method.”, authors suggested a hybrid method, named the CNN-LSTM-DNN, which is a combination of Convolutional Neural Network (CNN), Long Short Term Memory (LSTM), and Deep Neural Networks (DNN), for the estimation of the battery’s remaining useful life (RUL) and improving prediction accuracy with acceptable execution time. Their results revealed that hybrid methods perform better than the single ones, also the effectiveness of the suggested method in reducing the prediction error and in achieving better RUL prediction performance compared to the other methods.

In “A novel deep learning framework for state of health estimation of lithiumion battery”, the proposed approach is based on a hybrid neural network called gate recurrent unit-convolutional neural network (GRU-CNN), which can learn the shared information and time dependencies of the charging curve with deep learning technology. Then the SOH could be estimated with the new observed charging curves such as voltage, current and temperature. The approach is demonstrated on the public NASA Randomized Battery Usage dataset and Oxford Battery Degradation dataset, and the maximum estimation error is limited to within 4.3%, thus proving its effectiveness.

“A LSTM-RNN method for the lithium-ion battery remaining useful life prediction” investigates deep-learning-enabled battery RUL prediction. The long short-term memory (LSTM) recurrent neural network (RNN) is employed to learn the capacity degradation trajectories of lithium-ion batteries.

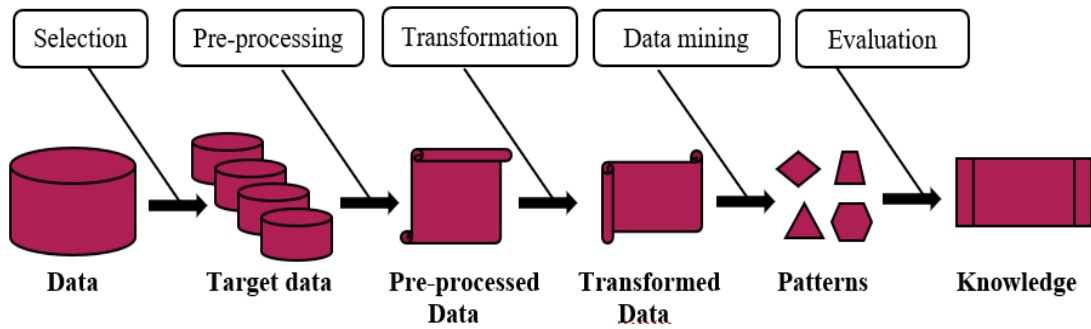
In the paper “State of health prediction of lithium-ion batteries based on machine learning: Advances and perspectives” an exhaustive comparison is conducted to elaborate the development of ML-based SOH prediction techniques. Not only their advantages and disadvantages of the application in SOH prediction are reviewed but also their accuracy and execution process are fully discussed.

In “Batteries State of Health Estimation via Efficient Neural Networks With Multiple Channel Charging Profiles”, adaptive boosting (AB) and support vector regression (SVR) are widely compared with long short-term memory (LSTM), multi-layer perceptron (MLP), bi-directional LSTM (BiLSTM), and convolutional neural network (CNN) to attain the appropriate approach for battery capacity and state of health (SOH) estimation. The research proves that BiLSTM outperforms all the approaches and obtains the smallest error values for MAE, MSE, and RMSE.

“State of Health Estimation of Lithium-Ion Battery Using Time Convolution Memory Neural Network” researches on how to extract battery’s features effectively and improve SOH estimation accuracy. Also, it proposes a time convolution memory neural network (TCMNN), combining convolutional neural networks (CNN) and long short-term memory (LSTM) by dropout regularization-based fully connected layer.

3. Methodology

KDD is the methodology used for this project. KDD stands for Knowledge Discovery in Databases. It is an iterative multi-stage process for extracting useful, non-trivial information from large databases. KDD methodology provides a structured approach for planning a data mining project.



3.1 Data Selection

The data that will be used for the knowledge discovery process should be determined once the objectives have been identified. The aspects that will be evaluated for the process include identifying what data is available, acquiring important data, and then integrating all of the data for knowledge discovery into one set.

3.2 Data Pre-processing

Data dependability is improved in this step. It includes data cleaning, such as dealing with missing values and removing noise or outliers. In this case, it might apply complicated statistical techniques or a Data Mining algorithm.

3.3 Data Transformation

This stage involves preparing and developing appropriate data for Data Mining. Dimension reduction and attribute transformation are two techniques used here. This stage is often project-specific and can be critical to the success of the entire KDD project.

3.4 Data mining

Having the technique, we now decide on the strategies. This stage incorporates choosing a particular technique to be used for searching patterns that include multiple inducers.

3.5 Evaluation

We examine and interpret the mined patterns, rules, and reliability in relation to the characterised objective in this step. This step focuses on the induced model's readability

and utility. The recognised knowledge is also documented in this phase for future use. The utilisation, as well as overall feedback and discovery outcomes obtained by Data Mining, is the final phase.

4. Design Specification

This project is implemented on Amazon EC2 G4dn.8xlarge server. (1 GPU, 128 GB Ram, 50 GBPS bandwidth). They are the lowest cost GPU-based instances in the cloud which is well known in the industry for implementing neural networks on real-time big volume data. Jupyter notebook installed in the server is used for coding equipped with a virtual machine running with a GPU processor. The hardware specification for this cloud platform varies from project to projects. In this project, it was implemented on a 128 GB RAM with 120 GB drive space and GPU support. GPU was selected as the runtime type for faster execution.

5. Implementation

5.1 Data Selection

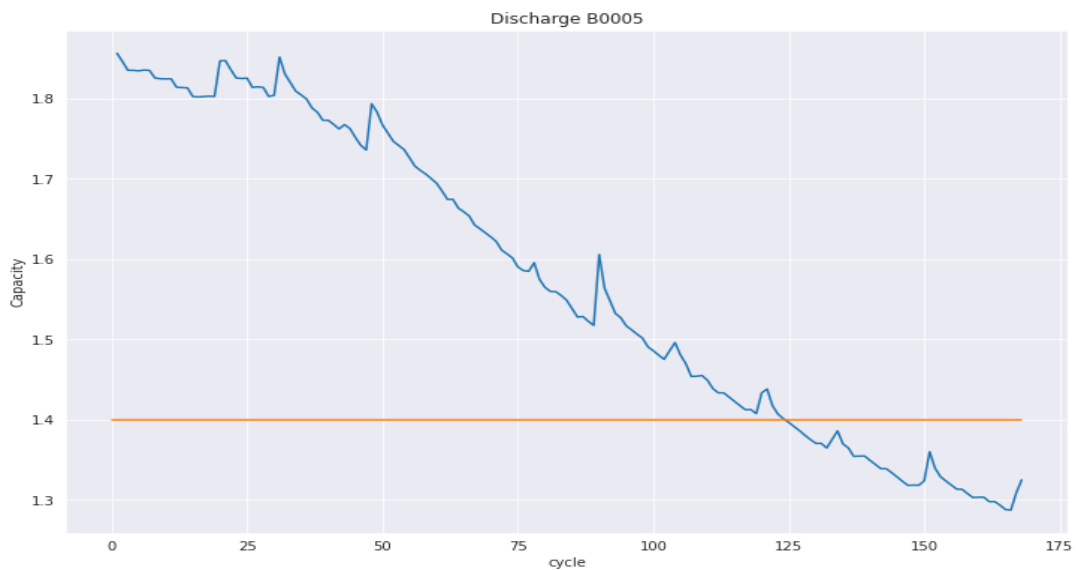
This research uses the NASA Prognostic Centre battery dataset. This dataset has four batteries reading B005, B006, B007 and B0018. These batteries contain three operational profiles (charge, discharge and impedance) at room temperature. Charging was carried out in a constant current (CC) mode at 1.5A until the battery voltage reached 4.2V and then continued in a constant voltage (CV) mode until the charge current dropped to 20mA. Discharge was carried out at a constant current (CC) level of 2A until the battery voltage fell to 2.7V, 2.5V, 2.2V and 2.5V for batteries 5 6 7 and 18 respectively. Impedance measurement was carried out through an electrochemical impedance spectroscopy (EIS) frequency sweep from 0.1Hz to 5kHz. Repeated charge and discharge cycles result in accelerated aging of the batteries while impedance measurements provide insight into the internal battery parameters that change as aging progresses. The experiments were stopped when the batteries reached end-of-life (EOL) criteria, which was a 30% fade in rated capacity (from 2Ahr to 1.4Ahr).

The data files are in MATLAB format which are converted into JSON format for better readability and understandability in python.

5.2 Exploratory Data Analysis

The crucial process of conducting an initial study on any given data is known as exploratory data analysis. With the use of statistical and graphical depiction, it is done to uncover anomalies, discover underlying patterns, test hypotheses, and check assumptions. Before moving on to the next level of data mining, it's always a good idea to first grasp the dataset.

In this project, since capacity is an important attribute and also involves in the process of prediction, it is plotted against the cycle number to see how it varies across cycles. The following graph shows the aging process of the battery as the charge cycles progress. The horizontal line represents the threshold related to what can be considered the end of the battery's life cycle.



Even though the threshold capacity is 1.4Ahr, the actual data goes beyond the threshold level and reaches around 1.3 Ahr in the last cycles.

5.3 Data preprocessing

5.3.1 Removing unwanted columns

Basic idea behind this step is to remove unwanted columns which can add noise to the

data. It was found that some columns in the data had less than 0.5 correlation towards the dependent variable. This means that these columns don't contribute much to the dependent variable and if kept, can increase unwanted noise to the model.

After removing unnecessary columns, attributes corresponding to the temperature, voltage and current remains.

5.3.2 Cleaning data

Interesting, it was found that around 8% of the total data is duplicated. To avoid miscalculations, when there exists more than one record for a particular instance, mean of all the column values are taken into account. All the duplicate entries are then removed from the dataset.

5.3.3 Adding columns

- Composite parameters were generated and added to the dataset.
Average SOC: $(SOC1 + SOC2 + SOC3 + \dots + SOC8)/8$
Total Voltage: $Vch1 + Vch2 + Vch3 + \dots + Vch8$
Average Current: $(Ich1 + Ich2 + Ich3 + \dots + Ich8)/8$
Average Temperature: $(T1 + T2 + T3 + \dots + T8)/8$
- State of health (SoH) was calculated from the capacity attribute. The formula used for the calculation is:
$$SoH = \frac{\text{Maximum battery capacity}}{\text{Rated battery capacity}}$$
- Time interval between each observation is calculated and is added as a new column in the dataset.
- Dataset is modified in such a way that it can be used by Tensorflow in the training phase, for this, two structures are created corresponding to the input and output expected to be obtained.

For the input data, the relevant characteristics of the dataset are filtered, which are:

Battery capacity
Voltage
Current
Temperature
Charging voltage
Charging current
Instant of time

For the output data, the SoH of the battery is calculated and in both input and output cases, the values are normalized to a range of values between [0-1].

5.4 Data transformation

When the time gap between each observation was analysed, it was found that the intervals are not constant which makes the time series data non-uniform. Typically, it ranges from 10 seconds to 20 seconds. Interestingly there also occurs a time gap of 2 days and more between 2 consecutive observations. There is a need to convert the nonuniform time series data to uniform time series. So, the first step is to give charge-discharge cycle numbers to the data observations.

After adding cycle number to the dataset, number of observations were made equal in all the cycles thus making the data uniform in nature

5.5 Data Mining

5.5.1 Time Series Decomposition

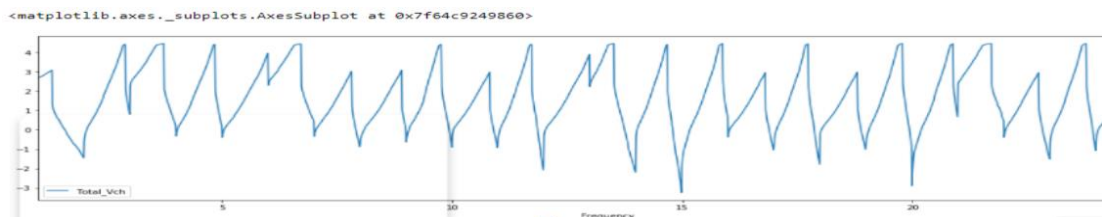
Decomposing a time series entails considering it as a collection of level, trend, seasonality, and noise components. Decomposition is a useful abstract paradigm for thinking about time series in general, as well as for better comprehending challenges encountered during time series analysis and forecasting.

Seasonality: describes the periodic signal in your time series.

Trend: describes whether the time series is decreasing, constant, or increasing over time.

Noise: describes what remains behind the separation of seasonality and trend from the time series.

The trend in total voltage across the cycles is shown below.



The graph clearly states that there doesn't exist any trend (neither upward nor downward trend) in the voltage data.

This is a clear indication that the data is stationary in nature.

5.5.2 Tests for stationarity

5.5.2.1 Augmented Dickey–Fuller test (ADF)

Augmented Dickey Fuller test (ADF Test) is a common statistical test used to test whether a given Time series is stationary or not. It is one of the most commonly used statistical test when it comes to analysing the stationary of a series.

Null Hypothesis (H 0): The series is not stationary

p-value > 0.05

Alternate Hypothesis (H 1): The series is stationary

p-value ≤ 0.05

After performing ADF test in our data, the p-value obtained is 0.00 which rejects the null hypothesis. Thus, it proves that the time series data that we have is stationary in nature.

5.5.2.2 Kwiatkowski-Phillips-Schmidt-Shin test(KPSS)

The KPSS test, short for, Kwiatkowski-Phillips-Schmidt-Shin (KPSS), is a type of Unit root test that tests for the stationarity of a given series around a deterministic trend.

In other words, the test is somewhat similar in spirit with the ADF test.

Null Hypothesis (H 0): The series is stationary

p-value > 0.05

Alternate Hypothesis (H 1): The series is not stationary

p-value ≤ 0.05

After performing KPSS test in our data, the p-value obtained is 0.1 which accepts the null hypothesis. Thus, it proves that the time series data that we have is stationary in nature.

5.6 Modelling

5.6.1 Recurrent neural network

Recurrent Neural Network is a generalization of feedforward neural network that has an internal memory. RNN is recurrent in nature as it performs the same function for every input of data while the output of the current input depends on the past one computation. After producing the output, it is copied and sent back into the recurrent network. For making a decision, it considers the current input and the output that it has learned from the previous input.

Unlike feedforward neural networks, RNNs can use their internal state (memory) to process sequences of inputs. This makes them applicable to tasks such as unsegmented, connected handwriting recognition or speech recognition. In other neural networks, all

the inputs are independent of each other. But in RNN, all the inputs are related to each other.

5.6.2 Transfer learning

Transfer learning for deep neural networks is the process of first training a base network on a source dataset, and then transferring the learned features (the network's weights) to a second network to be trained on a target dataset.

5.7 Evaluation

Evaluation is one of the most important part of any data mining project. This tells how good a model is performing in terms of different metrics. The following evaluation methods are used in this project to evaluate the performance of the model.

5.7.1 Root Mean Square Error (RMSE)

RMSE is the Root of the Mean of the Square of Errors and MAE is the Mean of Absolute value of Errors. Here, errors are the differences between the predicted values (values predicted by our regression model) and the actual values of a variable. They are calculated as follows:

$$\text{RMSE} = \sqrt{\frac{\sum (y_i - y_p)^2}{n}}$$

$$\text{MAE} = \frac{|(y_i - y_p)|}{n}$$

y_i = actual value

y_p = predicted value

n = number of observations/rows

5.7.2 R squared (R2 score)

R2 score is a metric that tells the performance of your model, not the loss in an absolute sense that how many wells did your model perform.

In contrast, MAE and MSE depend on the context as we have seen whereas the R2 score is independent of context.

So, with help of R squared we have a baseline model to compare a model which none of the other metrics provides. The same we have in classification problems which we call a threshold which is fixed at 0.5. So basically R2 squared calculates how must regression line is better than a mean line.

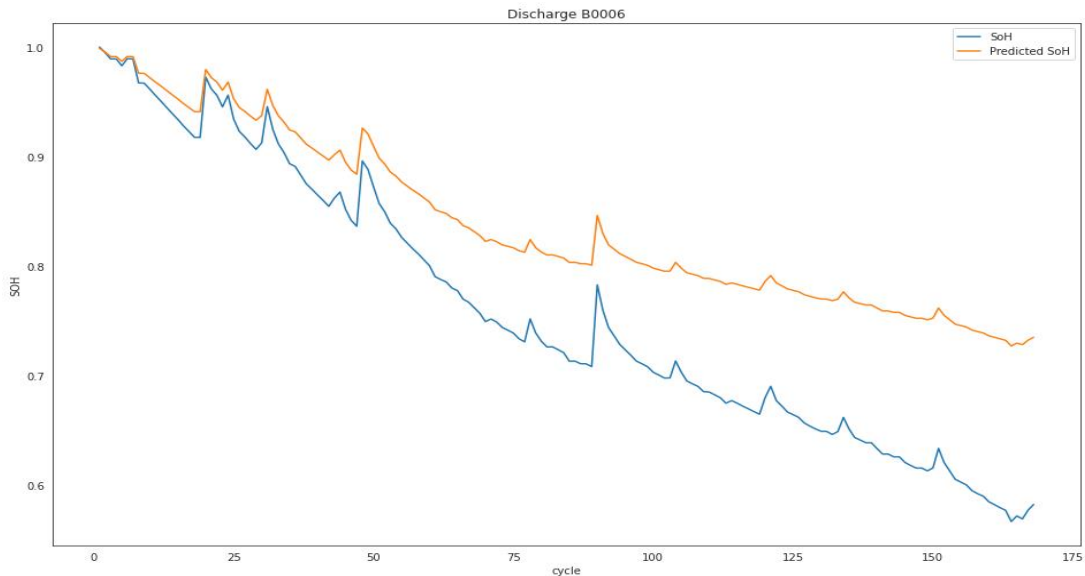
Hence, R2 squared is also known as Coefficient of Determination or sometimes also known as Goodness of fit.

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

6. Results

6.1 Prediction of State of health of battery

For the estimation of SoH, it can be seen that the data pattern is learned by the model correctly, as predicted by the theory, since the shape of the curves is almost identical. Root mean squared error obtained is 0.09.



6.2 Prediction of remaining useful life in battery

6.2.1 Base model: LSTM-RNN model

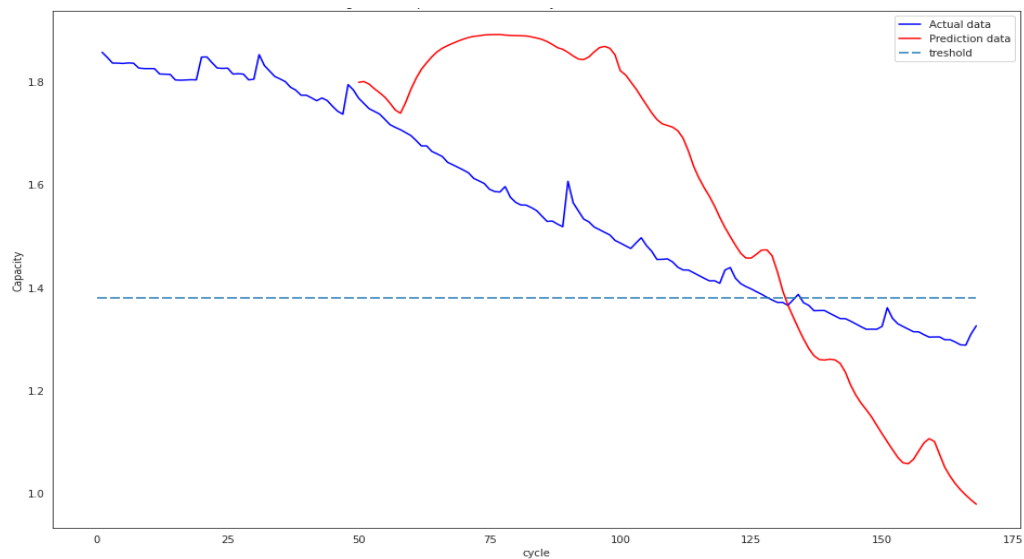
After training on B0005 brick, the model is used to predict the RUL in B0006. In the below figure, blue line shows the actual data and the red line shows the predicted data.

The Actual fail is at cycle number: 128

The prediction fail is at cycle number: 131

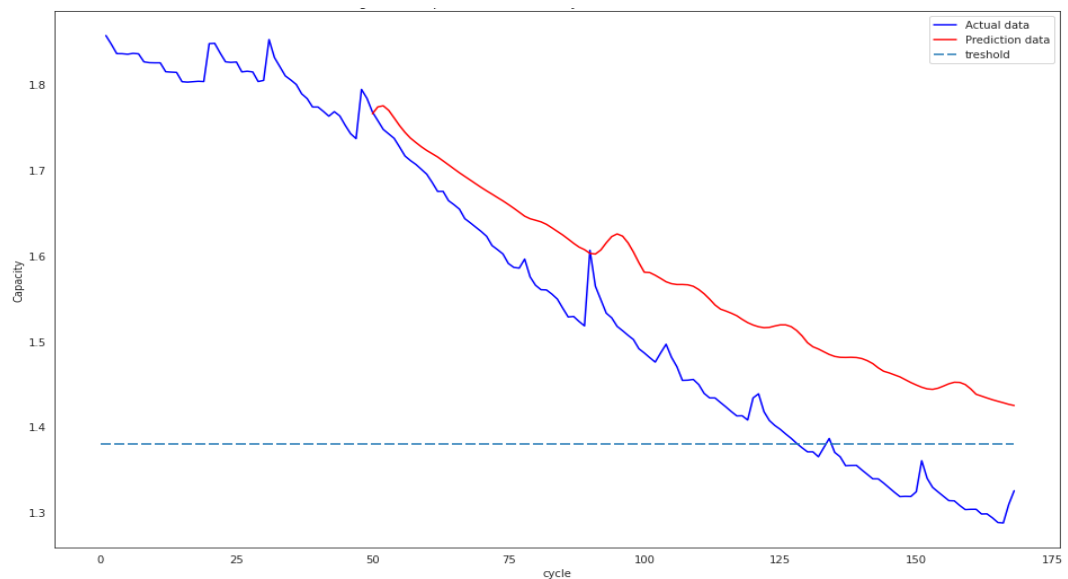
The error of RUL= 3 Cycle(s)

From the graph it is clear that the predicted RUL flow differs greatly from the actual flow. So, it cannot be considered as the final model. But can be used as a base model since the error is less as 3 cycles.



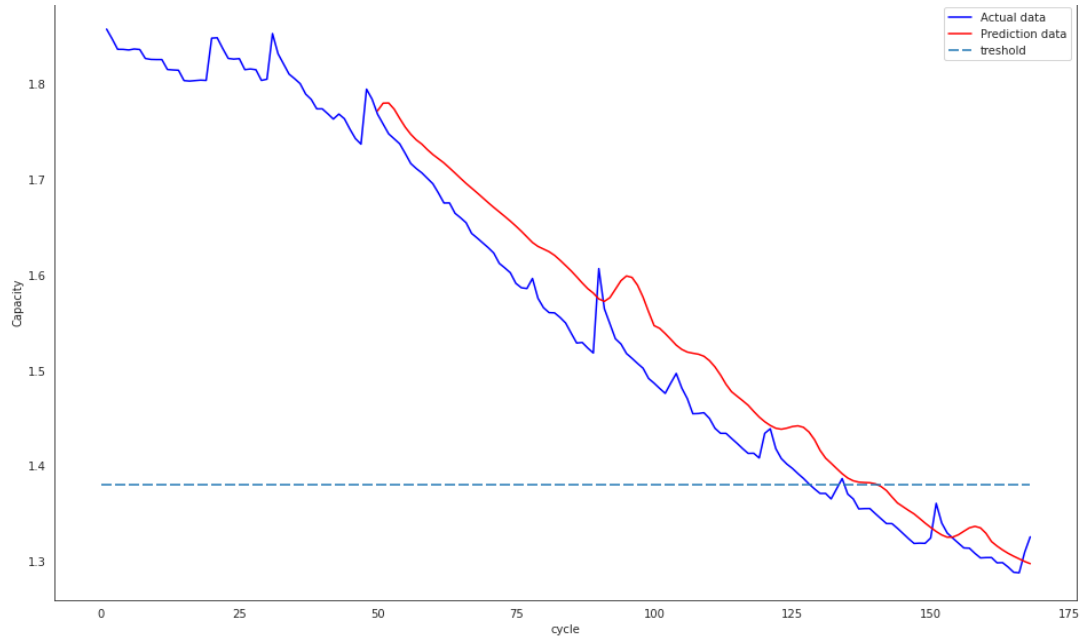
6.2.2 Transfer learning: B0007

The weights of the base model is passed on to the new model and is tested on the B0006. The model is now improved and the RUL flow is better compared to the base model. The error is 13 cycles



6.2.3 Transfer learning: B0017

After fitting the model on B0017, the path is much similar to the actual data and the error is also comparatively low (5 cycles).



7. Discussion

Some of the major challenges faced during the project was RAM being used. In order to overcome the same, various server machines like AWS server instances: t2.2xlarge (32GB) and m5.4xlarge(64GB) were tried.

The credibility of this project can be questionable, as the model only used a single type of brick to train and test the data. To tackle this problem, transfer learning is performed to attain the results from all the 3 different battery sets available. This would increase the credibility of the project.

8. Conclusion

This research was started with the aim of making an accurate prediction for RUL of Li-ion batteries, which will make the battery health monitoring more reliable and intelligent. This will help the electrical automobile industry, renewable energy plants etc. to maintain environmental sustainability and progress towards the clean energy goal.

Almost all the objectives from this research are fulfilled. This research proves that it is possible to recognise, monitor and analyse the voltage, current and temperature.

Dataset used, has only one ambient temperature, so to make model more robust it needs to be trained on dataset of other ambient temperatures. This research obtained the prediction only for the single cell of battery, while in real application usually a set of cells are used. Hence, there is a room to further extend this methodology to predict the RUL of entire battery pack.

9. References

- Y. Z. Zhang, R. Xiong, H. W He, and W. X. Shen, “Lithium-ion battery pack state of charge and state of energy estimation algorithms using a hardware-in-the-loop validation,” *IEEE Transactions on Power Electronics*, vol. 32, no. 6, pp. 4421–4431, Jun. 2017.
- Patil, M. A., Tagade, P., Hariharan, K. S., Kolake, S. M., Song, T., Yeo, T. and Doo, S, “A novel multistage support vector machine based approach for li ion battery remaining useful life estimation, *Applied energy*”, 159: 285–297, 2015.
- F. Sun, R. Xiong, and H. He, “A systematic state-of-charge estimation framework for multi-cell battery pack in electric vehicles using bias correction technique,” *Applied Energy*, vol. 162, pp. 1399-1409, Jan. 2016.
- A. Nuhic, T. Terzimehic, T. Soczka-Guth, M. Buchholz, and K. Dietmayer, “Health diagnosis and remaining useful life prognostics of lithium-ion batteries using data-driven methods,” *Journal of Power Sources*, vol. 239, pp.680–688, Oct. 2013.
- Zheng, X. and Fang, H, “An integrated unscented kalman filter and relevance vector regression approach for lithium-ion battery remaining useful life and short-term capacity prediction, *Reliability Engineering and System Safety*”, (2015).

