ANALYZING A NEW FEATURE FOR STATE OF CHARGE ESTIMATION OF LI-ION BATTERIES

by

DEVANG MARVANIA

THESIS

Submitted to the Graduate School

of Wayne State University,

Detroit, Michigan

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

2021

MAJOR: MECHANICAL ENGINEERING

Approved By:

Advisor Date

DEDICATION

This work is dedicated to a lot of people, without whom I won't be able to reach here. Special thanks to, my family, teachers and mentors - for their infinite patience, support and trust.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to Dr. Leela Arava. His positive attitude is what kept me going when going got tough. I was able to explore my interests under his wings and he supported me at every step of the way. Also, I would like to thank my committee members - Dr. Mahbub Islam and Dr. Golam Newaz for their interest in my work and forming my thesis committee.

Dr. Ephrem Chemali's PhD work was my inspiration to explore data driven approaches for the battery state estimations. He provided a great detail about experimental setup of his work and open sourced the data which is backbone of this work. Very thankful for his contributions and help.

The journey to complete masters while working full time would not have been possible without close family and friends. I would like to express my deepest gratitude to my parents, for motivating me to work harder and aim bigger. I am thankful to my brother and sister in law for always supporting me and cheering me up. I am always indebted to my wife, for her huge sacrifice and putting my needs over her. I am thankful to my brother in law, for helping with me during machine learning model developments.

Additionally, I would like to the acknowledge the importance of luck in this achievement - it played in my favor tremendously, just like all other good things happened to me.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

1.1 Introduction

Recently Li-ion batteries have become of significant commercial and research interest due to the key role they play in powering various technologies today. Traditionally Li-ion batteries have been used in consumer electronics such as laptops and smartpho-nes, power tools, and other industrial applications. The main drivers for growth in commercial demand are increased production of electrified vehicles and grid storage solutions. The global Li-ion battery market size was valued at USD 32.9 billion in 2019. It is expected to grow at a compound annual rate of 16% with a 2023 valuation of USD 62 billion[1].

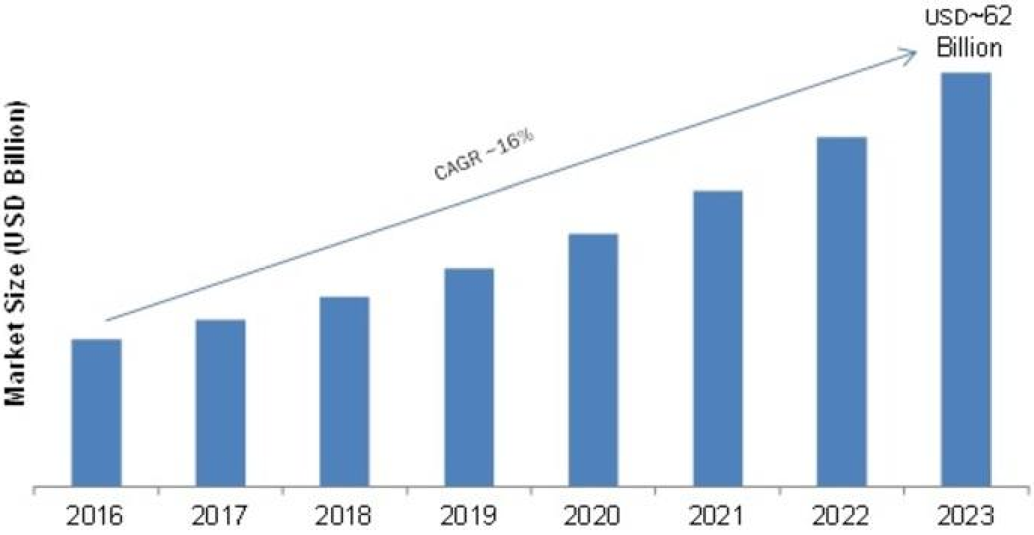


Figure 1.1: Growth of worldwide Li-ion Battery Market

Currently, the automotive market is shifting to implement more and more electrificati-on in vehicles. This trend is the result of many factors:

(i) Tightening regulations for tailpipe emissions[2]: Automakers have been steadily working towards tailpipe emission reduction and increment of fuel efficiency to comply with regulations[3]. To comply with future regulations, electrification has become a need rather than a choice. Electrification becomes ever more attractive when we consider higher efficiencies of electrified powertrains[4,5] and cost savings realized by using off-peak hour electricity[6,7].

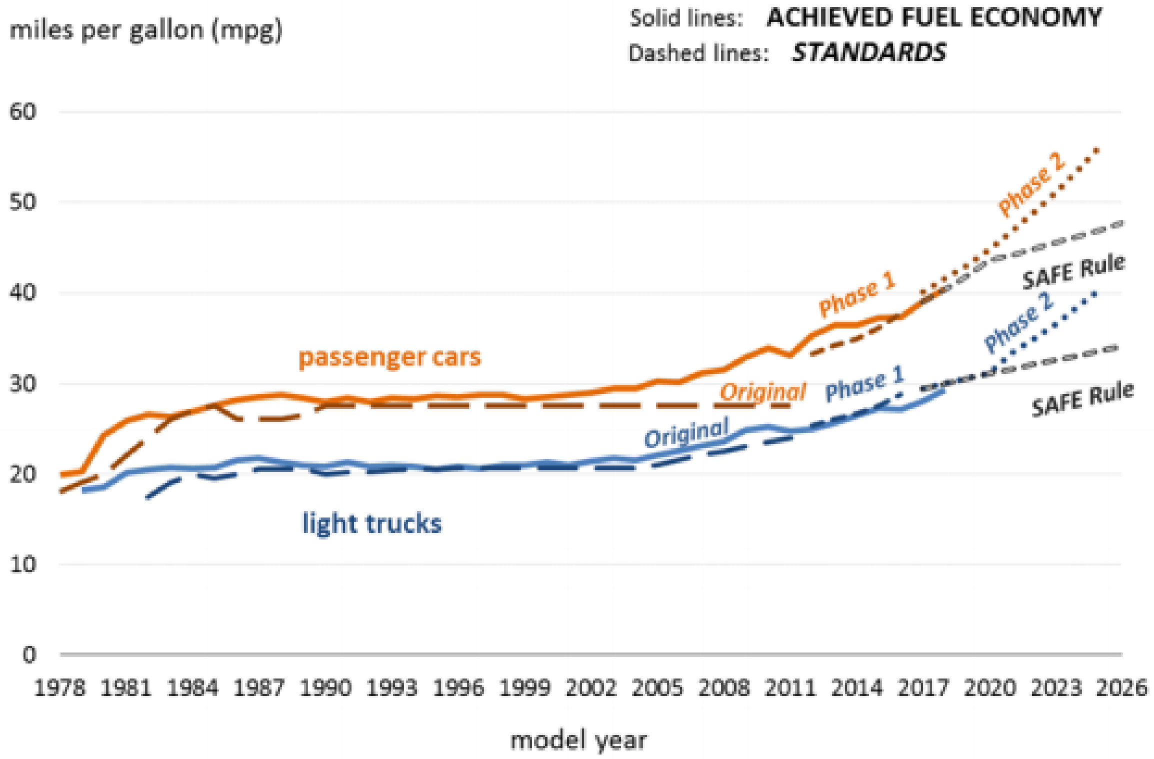


Figure 1.2: Increasing CAFE standards for Fuel Efficiency

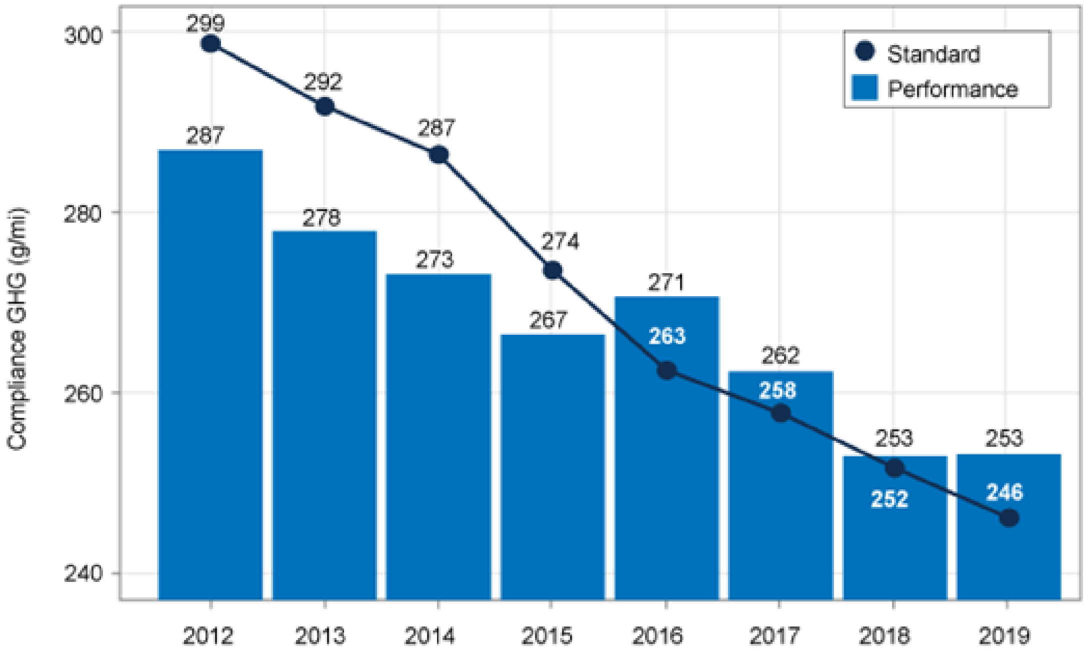


Figure 1.3: Decrease in allowable GHG emissions

(ii) Volatile Fossil fuel prices: Coal, Gasoline, and other fossil fuel prices tend to be volatile[8]. Also, their supply and demand balance can easily be disturbed by political tensions between producer and consumer countries. For a long-term stable growth of a nation, it is imperative to have a reliable source of energy to power the nation's needs. Energy scarcity can negatively affect the nation's economy[9,10]. Many nations have and are still promoting the use of renewable energies, electric vehicles, and biofuels[11,12].

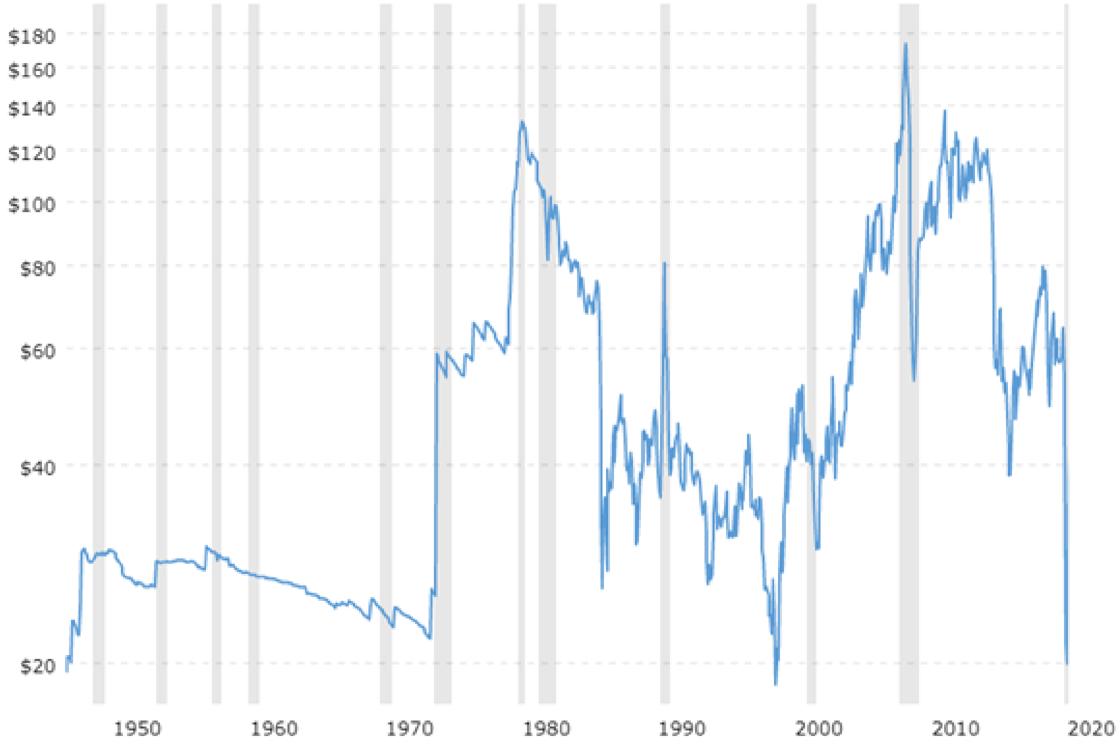


Figure 1.4: Instability in Crude Oil per barrel prices

(iii) Decreasing Battery and electric vehicle costs: In the past decade, Li-ion battery prices have been steadily decreasing making it attractive to integrate the battery in traditional vehicle powertrains[13]. This gives rise to various electrified powertrains: Hybrid, Plug-In Hybrid, Range Extender Electric Vehicle, and Battery Electric Vehicle(B-EV). Falling prices of batteries put BEVs in the realm of traditional ICE-powered vehicles from the cost point of view.

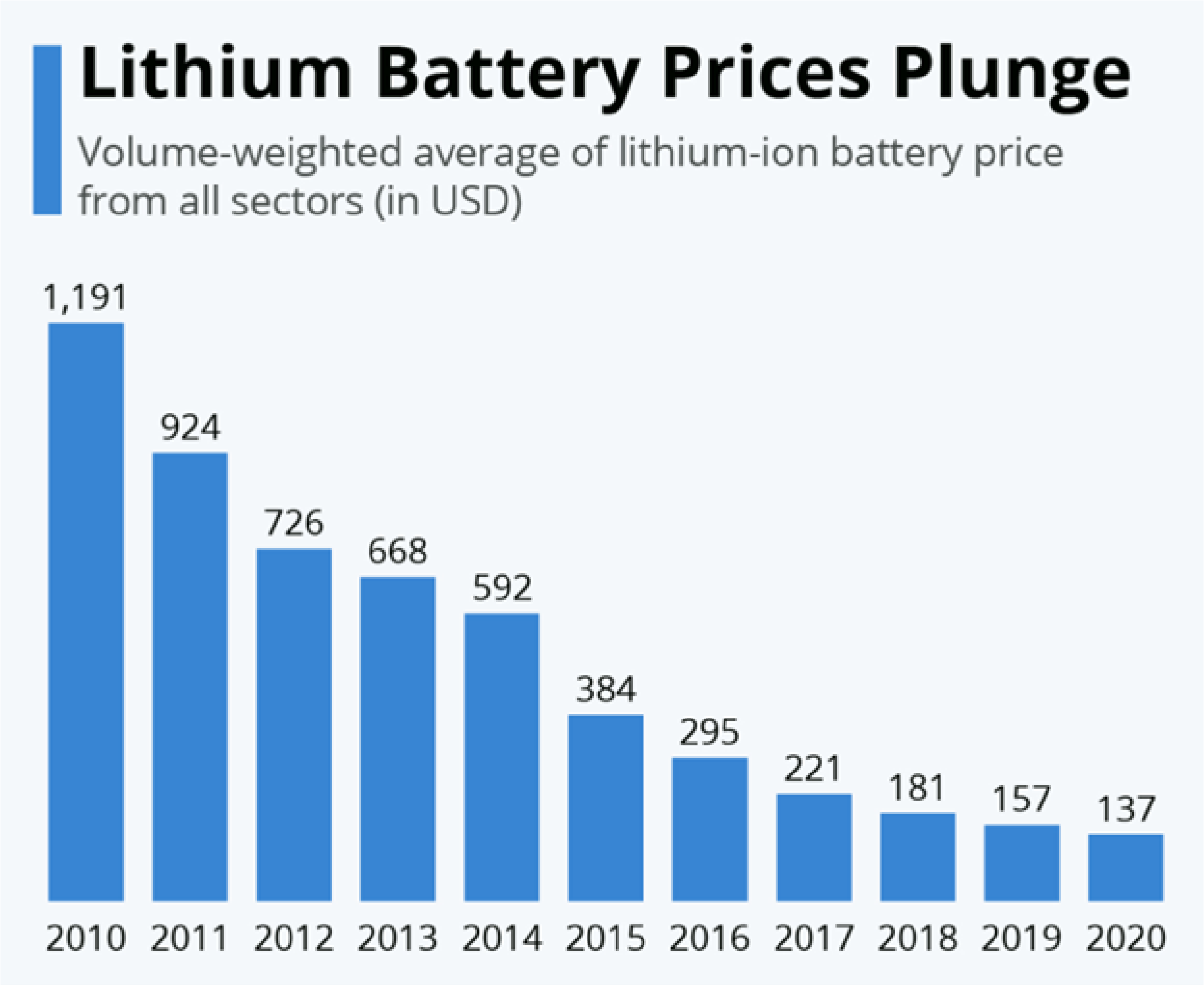


Figure 1.5: Falling Li-ion Battery prices[USD/kWh-hr]

Along with the above-mentioned factors, with the advent of Tesla, consumers are demanding exciting electric vehicles from manufacturers. As a result, more and more auto manufacturers are introducing Electrified models in their road maps. Some of the announcements from major manufacturers include [14] : (i) General Motors to launch 30 new global electric vehicles by 2025. (ii) Ford aims to have 40% of volume all-electric by 2030 (iii) Toyota to launch 15 new Battery Electric Vehicles by 2025.

With Li-ion batteries being the primary or secondary source of energy in these vehicles, it is imperative to have robust control over the performance and behavior of such systems. Li-ion batteries degrade over time and store less and less energy with time. Also, their performance depends heavily on ambient conditions and usage. These factors make Li-ion batteries complicated to model and control accurately.

Typical indicators of Li-ion battery are the State of Charge(SOC), State of Health(S-

OH), State of Power(SOP) etc. These indicators must be measured indirectly using some sort of estimator. The estimator uses physically measurable quantities to calculate these states. Typically, these physical quantities are the Temperature of the battery, Surrounding temperature, Current being charged/discharged, Terminal Voltage, etc.

1.2 Thesis Structure

Due to the demand for more reliable and robust battery technology, Battery Management Systems (BMS) algorithms are a prime research area. In this work, chapter 2 covers various methods currently being explored for State of Charge estimation including trending data-driven methods. Ch-3 provides the necessary theoretical background to understand a specific data-driven method: Long Short-Term Memory – Reinforced Neural Network (LSTM-RNN). Ch-4 explains the construction of two LSTM-RNN models which predict the state of charge of a Li-ion battery. Two categories of models are trained : 3 feature models and 4 feature models. 3 feature models use Voltage, Temperature, and Current to predict the SOC of Li-ion Battery, while 4 feature models use an additional feature: Voltage difference. Both categories of models are trained using various hyper-parameters and a comparative study of model performance is done. Ch-5 concludes the work assessing robustness of LSTM models to predict SOC, assessing if the additional feature of Voltage difference provided any extra help in SOC prediction and some ideas for future work directions.

CHAPTER 2 SOC ESTIMATION AND DATA DRIVEN APPROACH

2.1 State of Charge(SOC) estimation

Accurate real-time monitoring and control are critical to any application employing Li-ion batteries. This is done by the Battery Management System (BMS) – typically a chip embedded in the battery system [15]. SOC is a critical item to be monitored by BMS. SOC is important to understand the remaining amount of energy in the battery as well as for safe operation during charging and discharging. SOC is also an important parameter for proper balancing of the battery pack[16,17]. The accuracy of a SOC estimation method is quantified by calculating some form of error metric. Typical metrics include: (i) Mean Average Error (MAE) (ii) Root Mean Squared Error (RMSE) (iii) Maximum Error (ME). Error is calculated as the difference between the predicted value and ground truth SOC value.

2.2 SOC estimation methods

SOC estimation methods can be classified in following categories[18]: (i) Direct Measurements(ii) Bookkeeping Estimations(iii) Model-Based Methods(iv) Computer Intelligence.

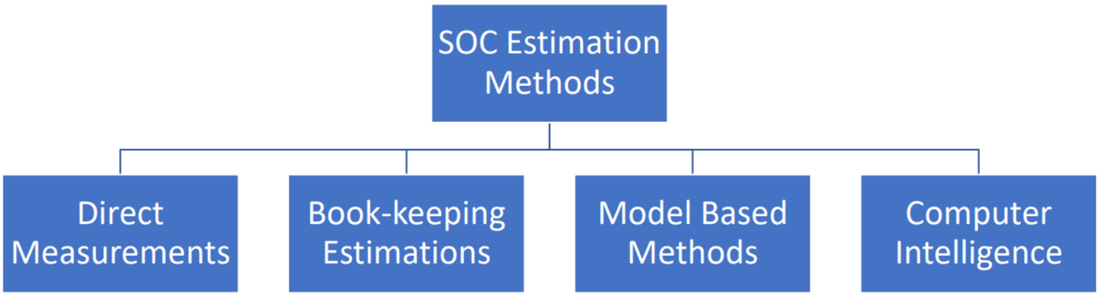


Figure 2.1: Classification of the SOC methods

2.2.1 Direct Measurements

In these methods of SOC estimation, a certain characteristic of the battery is measured and SOC is mapped to that measured element. For example, in OCV (Open Circuit Voltage) and EMF(Electro Motive Force) method: SOC is mapped with OCV/EMF of the battery[19,20]. For the Internal resistance method, the resistance of the battery is measured using current flowing through the battery and terminal voltage. During measurement of the internal resistance, reading can be affected by transfer reaction and acid diffusion effect. Typically, this is not a very accurate method. In Impedance Spectroscopy, the impedance of the battery is measured by applying various current frequencies. After getting impedance, SOC can be plotted using those impedance values.

2.2.2 Book-Keeping Measurements

As the name suggests, in this method a particular quantity is monitored and integrated over time to estimate SOC. The quantity being monitored is the charging or discharging current. SOC can be measured per equation below [18]:

$$SOC(t) = SOC(t_0) - \frac{1}{Q}\int_{t_0}^{t} (\eta i_{bat}(t) - s_d)\, dt$$

Here, η is the coulombic efficiency, i_{bat} is instantaneous charging or discharging current (positive for discharging current and negative for charging current), s_d is the self-discharge rate and Q is the nominal capacity. While the implementation is quite simple on a controller, it may skew the reading since sensor error in current measurement will propagate and continuously become bigger and bigger. Also, this method requires the initial SOC to be a known value.

2.2.3 Model-Based Methods

Direct measurements and Book-Keeping methods while relatively simple, do not offer much value in a practical sense as they cannot offer efficiency and high accuracy in real-time. Model-based methods tend to provide a good advantage on this aspect. Basic physical signals like Voltage, Current, and Temperature are used to model the battery. Then an error for Voltage is calculated by taking a difference between true and modeled value. This error when fed to an Observer, will predict the SOC Value. SOC value is fed back to the Battery model to make a closed loop. This way, the model is further tuned to reduce the voltage error and produce a more accurate SOC. The overall process flow is described in figure 2.2[18]. Some of the examples in this category are Luenberger Observer[19,21,22,23,24],Adaptive Observer[19,25,26],Sliding mode Observer[19,27,28,29-

,30,31,32,33] and Kalman filters[19,34,35,36,37,38,39].

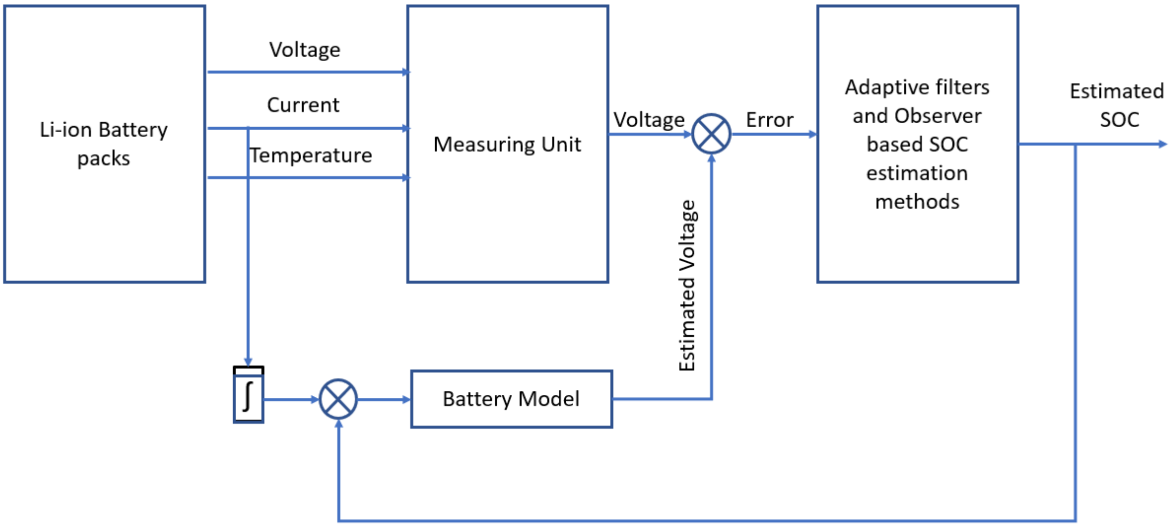


Figure 2.2: General diagram of model-based SOC estimation methods

2.2.4 Data-driven/Computer Intelligence approaches

With the advent of Artificial intelligence (AI), this is the latest addition to SOC estimation techniques. In this work, the focus will be on data-driven approaches. Artificial Intelligence is a broad field, and it can be represented as figure 2.3[40] in the context of Machine Learning and Deep Learning.

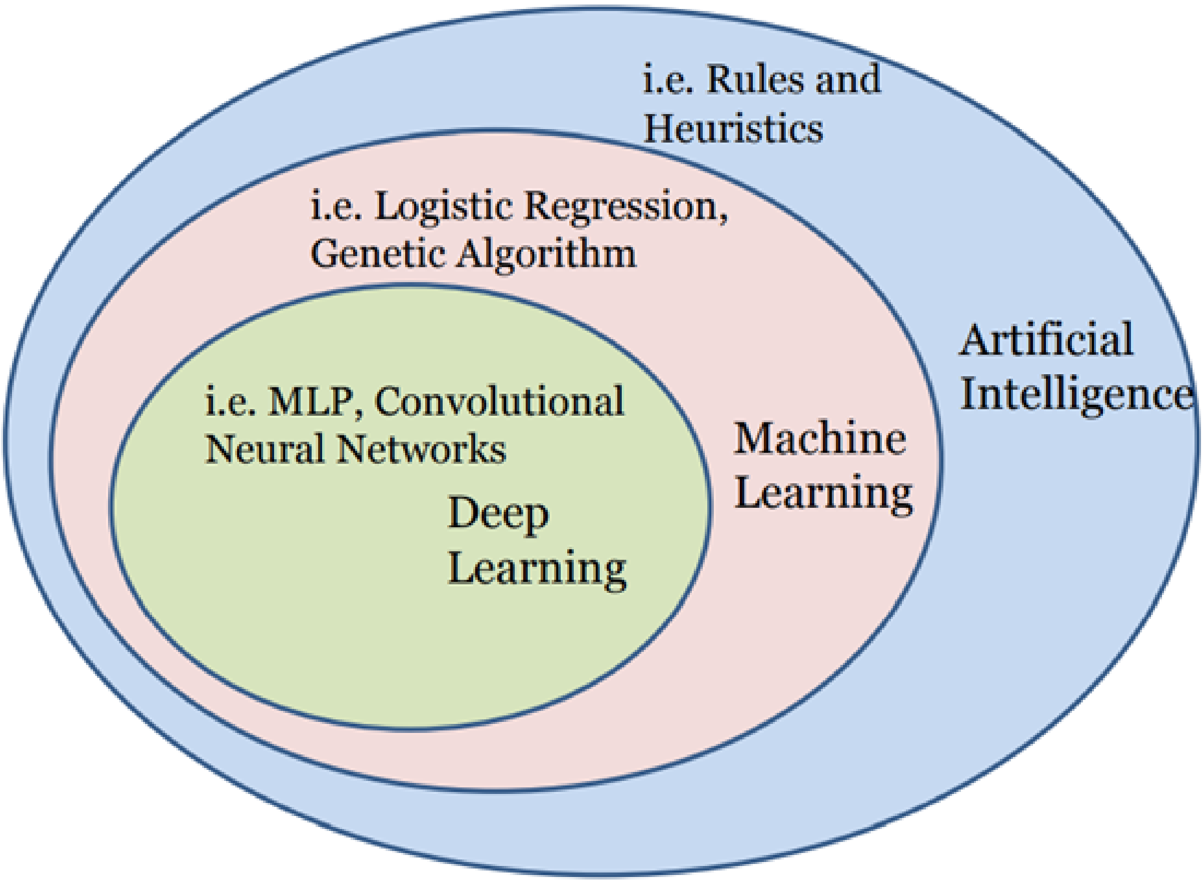

i.e. Rules and Heuristics
i.e. Logistic Regression, Genetic Algorithm
i.e. MLP, Convolutional Neural Networks
Deep Learning
Machine Learning
Artificial Intelligence

Figure 2.3: Venn diagram of the field of AI

Learning algorithms based on AI can perform a variety of tasks. Two common tasks are Classification and Regression. The classification task is assigning the category to an input of the learning algorithm. For example: If a cat image is fed to the model, the model should be able to predict whether the picture is of Cat or Non-Cat. The regression task involves a model to predict a numerical value by processing the inputs provided. For example: Given the humidity, temperature, wind direction, etc. predict the dew point of the air. The data-driven models for SOC estimation are of Regression type since the output is a numeric value- SOC.

Deep learning methods for battery modeling involve network creation and training of the network parameters. When network parameters are trained properly, they represent the battery's characteristics. To train the network parameters, network parameters are randomly initialized, and predictions are made using these initial parameters. When this network is given a set of inputs, it outputs the quantity of interest. Passing the raw input data from network inputs to outputs is called Forward propagation. During this process, inputs are multiplied with network parameters (weights and biases), passed through transformation functions called Activation functions to arrive at the final output. This process is known as Forward propagation. Now, the output is compared with ground truth values, and parameters are adjusted to reduce the error between predicted and true values. The adjusting process starts from the final layer of the network and moves back to the first layer. This way, the network learns the system dynamics by adjusting network parameters. This process is known as backward propagation.

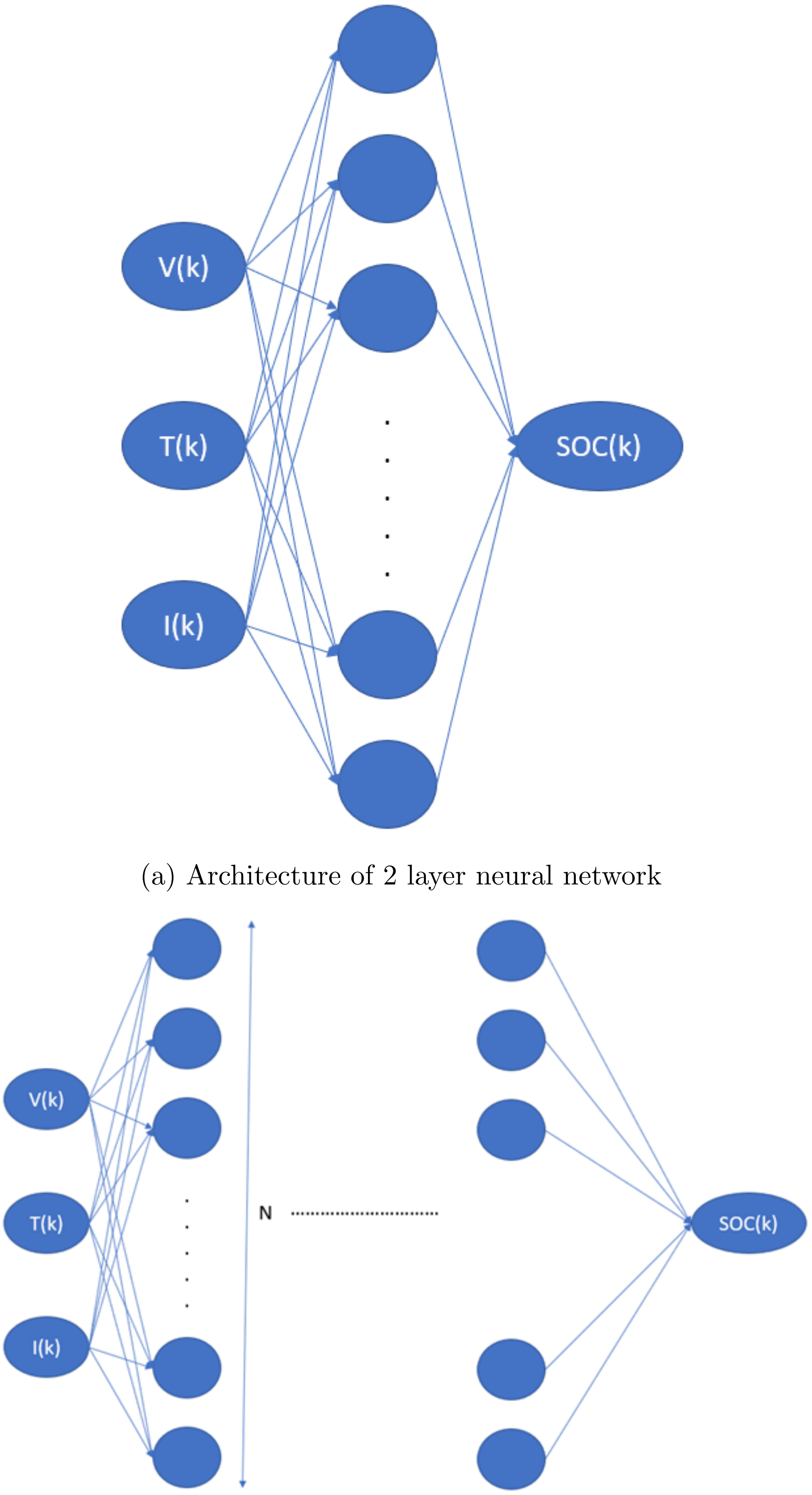
For SOC estimation, typically the input data used are Voltage, Current, and Temperature. These data streams are commonly measured signals in Battery Management Systems. Deep learning methods have certain advantages over other methods. Mainly, the number of network weights and biases can be changed to capture system complexity. Additionally, this approach is suitable to automate training with real-time data. This can be accomplished by sending a stream of data to a cloud-based server, training in the server, and getting trained model back from the server.

CHAPTER 3 LSTM-RNN MODEL FOR SOC PREDICTION

3.1 Neural Network Basics

Before diving into LSTM-RNN (Long Short-Term Memory – Reinforced Neural Network), it is useful to understand how a Neural Network works. Traditional neural networks have been 2-layer networks, but with advances in computational abilities and open-source data, deep architectures are becoming practical, which provides striking model accuracy improvements[40].

Feedforward neural networks with 2 layers and more than 2 layers are shown in the figure 3.1a and 3.1b. It implements non-linear mapping between given input and output.



(a) Architecture of 2 layer neural network

(b) Architecture of Deep Neural Network

Figure 3.1: 2 layer and Deep Neural Networks

Typical input and output used in neural network for SOC estimation are X(t)=[Voltage(t), Current(t), Temperature(t)] and y(t)=[SOC(t)] respectively. Here, y(t) is the ground truth value of SOC which will be used to measure the accuracy of the predicted value once the model is trained. Input is linked to neurons via network parameters called weights(w) and biases(b). For a fully connected network, increase in the number of weights due to additional inputs = (additional inputs) x (number of neurons in the first layer).

The effect of input on the output of a neuron is given by the associated weights and biases attached to the connection between that input and neuron. All inputs are multiplied by their respective weights and summed, later a bias term is added to that sum. The final value of the sum is z as shown in the figure 3.2[41]. This z becomes an argument for the activation function g. Graphically, the inner workings of a neuron can be represented per figure 3.2[41].

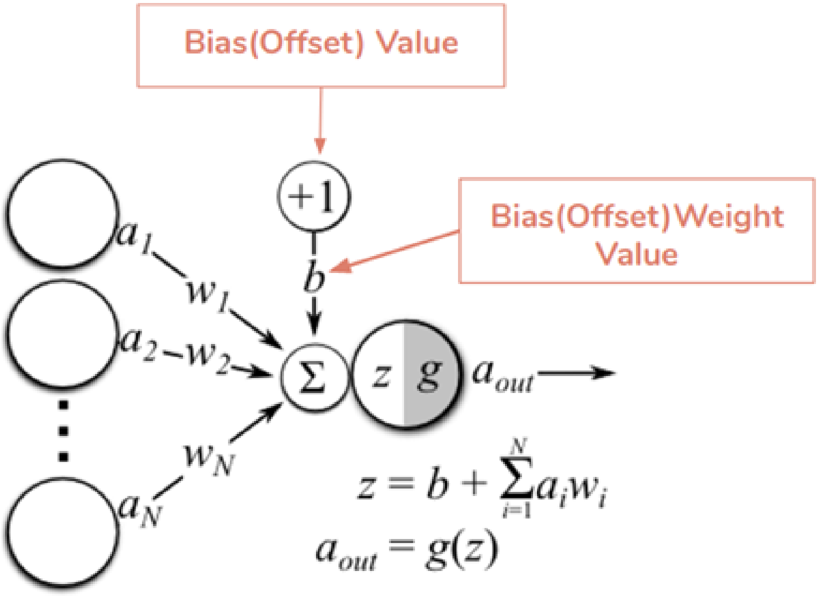


Figure 3.2: Output of a neuron with given inputs

The activation function introduces non-linearity in a neural network. There are a variety of activation functions available but commonly used are sigmoid, tanh, and Rectified Linear activation functions.

Sigmoid activation:-

$$g(z) = \frac{1}{1 + e^{-z}}$$

Sigmoid activation can be plotted as figure 3.3[42].

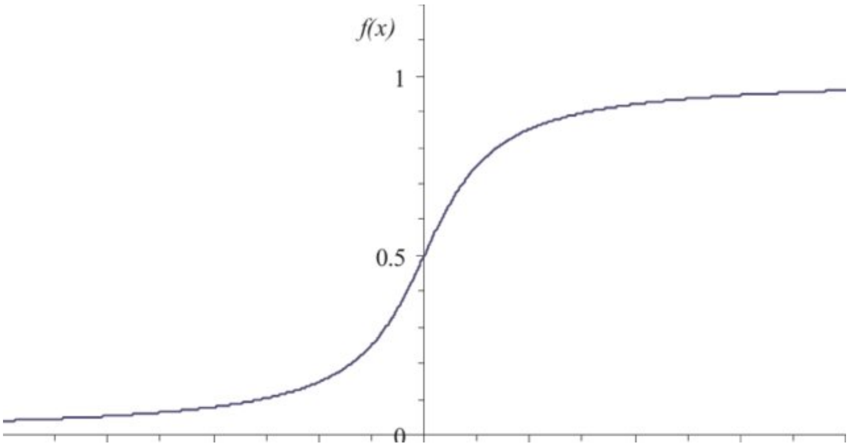


Figure 3.3: Sigmoid activation function plot

Hyperbolic Tangent activation:-

$$g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

Hyperbolic tangent activation can be plotted as figure 3.4[43].

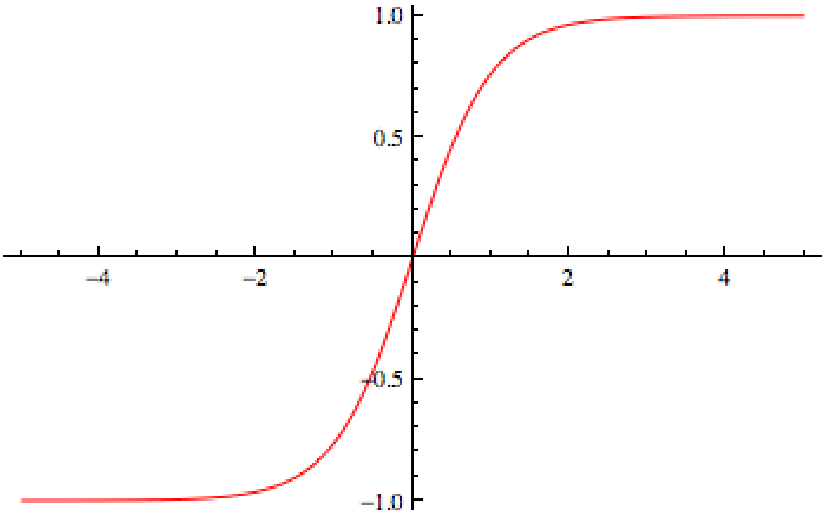


Figure 3.4: tanh activation function plot

Rectified Linear Unit(ReLU) activation function:-

$$g(z) = max(0, z)$$

ReLU activation can be plotted as figure 3.5[44].

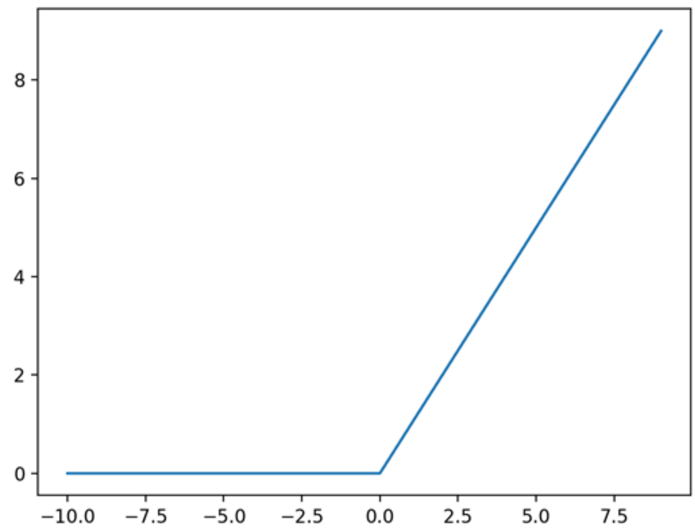


Figure 3.5: ReLU activation function plot

In a neural network, once the output of the first layer is calculated, they propagate further in the network by multiplying with respective weights, added by biases, and transforming through activation functions till it reaches the end of the network. This activity is called a forward pass.

Here, the output of the network is compared with the ground truth values of a given input, and error is calculated. For, a NN which predicts SOC, one example of error would be

$$e(t) = SOC(t) - SOC^{*}(t)$$

In the equation above, SOC*(t) is the predicted SOC value by the neural network and SOC(t) is the true value of SOC. Based on the error, a loss function is calculated. Loss function can be defined in multiple ways, some of the common loss functions are: Mean Absolute Error, Root Mean Squared Error, and Max Error.

Mean Absolute Error Loss Function: -

$$L = \frac{1}{n}|SOC(t) - SOC^{*}(t)|$$

Root Mean Squared Error Loss Function: -

$$L = \sqrt{\frac{\sum_{i=1}^{n}(SOC(t) - SOC^{*}(t))^{2}}{n}}$$

Max Loss Error Function:-

$$L = max(|SOC(t) - SOC^{*}(t)|)$$

In the equations above, n indicates the total number of readings.

3.1.1 Forward and Backward Propagation:-

To generate accurate predictions from the model, it needs to be trained with training data. Training the model is basically updating its weights and biases while minimizing the loss function. One full training epoch is defined as one forward pass and one backward pass. In the forward pass, training data is fed into the network and the network produces prediction after passing through the last layer. In backward pass, the loss is calculated and passed through the network in reverse order – the last layer first. During the process, the gradient of loss is calculated with respect to network parameters. This gradient will be used to calculate the updated values of parameters. In this work, to update network parameters, an optimized method called "Adam" is used which can be given by the below composite function[45]:

$$u_{\epsilon} = \gamma_1 u_{\epsilon-1} \vec{\boldsymbol{\nabla}} L(w_{\epsilon-1})$$

$$v_{\epsilon} = \gamma_2 v_{\epsilon-1} \vec{\boldsymbol{\nabla}} L(w_{\epsilon-1})^2$$

$$\widetilde{u_{\epsilon}} = \frac{u_{\epsilon}}{1 - \gamma_1^{\epsilon}}$$

$$\widetilde{v_{\epsilon}} = \frac{v_{\epsilon}}{1 - \gamma_2^{\epsilon}}$$

$$w_{\epsilon} = w_{\epsilon-1} - \rho \frac{u_{\epsilon}}{\widetilde{v_{\epsilon}} - k}$$

Here, γ_1 and γ_2 are decay rates, ρ is the learning rate and k is a constant term. The network weights at the present training epoch are given by w_{ϵ}. A normally distributed random number generator is used to initialize the network weights.

3.2 Workings of LSTM-RNN

Reinforced Neural networks are the type of neural network that loop in the back. This arrangement allows them to remember the past information and use that remembered past information to make predictions in the current time. This way, they can remember history. Graphically, represented in Figure 3.6[46].

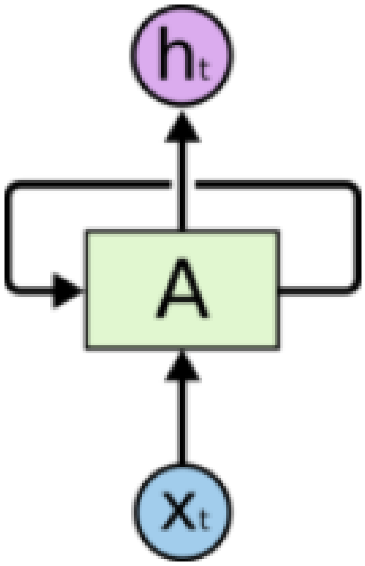


Figure 3.6: A single neuron with recurrent connection

In the figure3.6, a single neuron is shown which is part of a neural network. This neuron looks at the input given at time t, X_t and produces output h_t. Along with that, it also saves a portion of output which will be used later at time $t + 1$ with input X_{t+1}. If this network is unrolled in time, it may look like the figure 3.7[46].

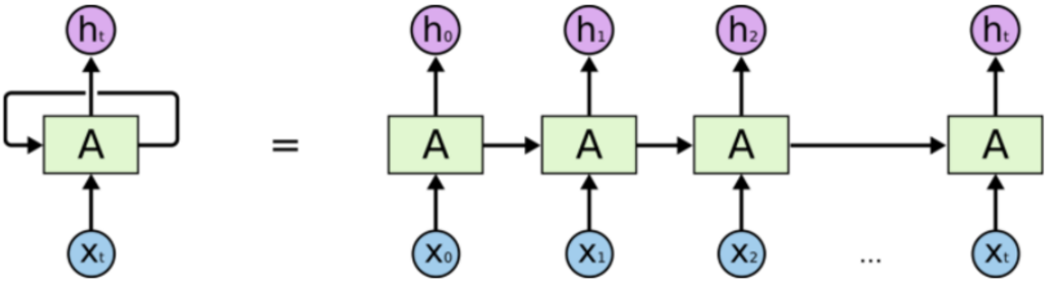


Figure 3.7: An unrolled recurrent neural network

This chain-like structure makes RNN very useful for applications where having knowledge of past information can be helpful in predictions. Some of the common applications are Speech recognition, Language modeling, Translation, Image captioning, etc[46]. Recently, Battery state estimations models are also leveraging RNNs since having past usage information helps in making predictions of the current state in Li-ion Battery. Despite their advantages, RNNs suffer from gradient vanishing and the inability to decide how much to remember past. To address these issues, a variant of RNN is often used. That is known as Long Short Term Memory – Reinforced Neural Network(LSTM-RNN).

LSTMs can be represented as chain-like structures like RNN, as shown in Figure 3.8[46].

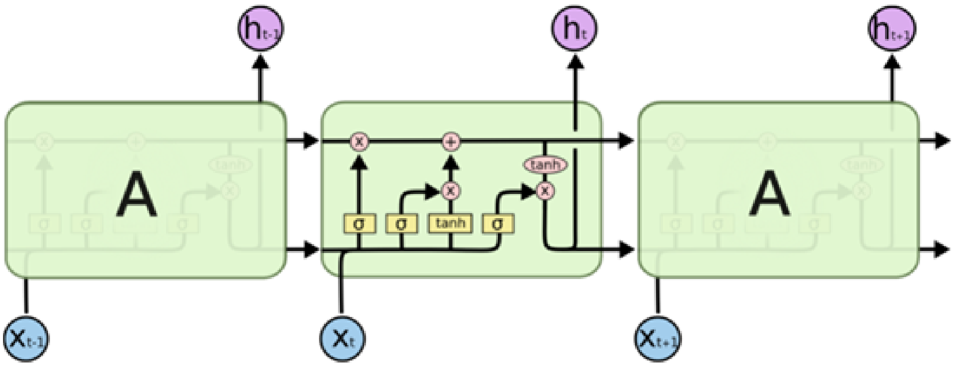


Figure 3.8: Interacting layers within an LSTM block

The main idea behind LSTM is the cell state, which is horizontal line running through the LSTM block in figure 3.9[46]. During operation, information is removed or added to the cell state. The mechanism through which information being added or removed is called ‘gates’. In below diagram, h_{t-1} = output of previous LSTM bock, C_{t-1} = cell state from previous LSTM block, x_t = input at time t, C_t = cell state of this LSTM, h_t = output of this LSTM block at time t.

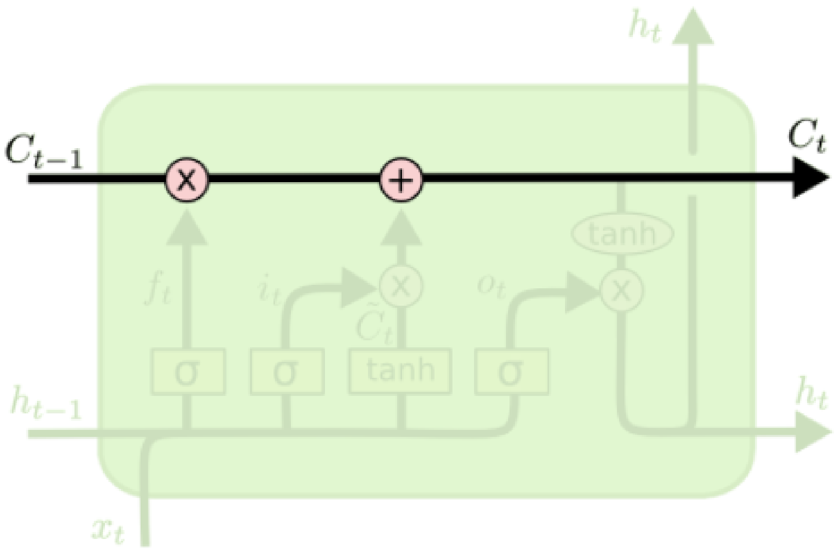


Figure 3.9: Cell state highlighted in an LSTM block

First step in the LSTM block is deciding what information to keep from the cell state. This action is done by a sigmoid layer called the 'forget' gate. It takes h_{t-1} and x_t as input and produces f_t.

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

Second, the LSTM block decides what information will be stored in the cell state. This happens in 2 parts: (i)A sigmoid layer called the "input" gate decides how much values will be updated. It takes h_{t-1} and x_t as input and produces i_t. (ii) A tanh layer creates a vector that could be added to the cell state. This new value vector is denoted by $\widetilde{C}_t$.

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

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

Now, Cell state is updated using the formula:

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C}_t$$

Third, after calculating cell state, cell need to produce an output h_t for the LSTM block. This happens in two parts : (i) A sigmoid layer called the "output" gate calculates how much cell state will be converted to output. It takes h_{t-1} and x_t as input and produces o_t. (ii) A tanh layer is applied to the Cell state to scale it between -1 and 1.

Final cell output is calculated as below:

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

$$h_t = o_t * tanh(C_t)$$

3.3 Scope of this work

In this work, two variations of LSTM-RNN models are generated. Both are designed to predict the SOC of the Li-ion Battery. The first variant has 3 input features: Voltage, Current, and Battery Surface Temperature. The second variant has 4 input features: Voltage Difference, Voltage, Current, and Battery Surface Temperature. An attempt is made to explore how having Voltage Difference as one of the features changes model performance. Having more features can affect model performance both ways: It can improve performance if new features provide additional information to better model the system or it can decrease or do not change significantly if two or more features convey the same intelligence to the model or if new feature introduces noise. To keep the comparison relevant, the hidden layer and number of LSTM and Dense units in hidden layers are kept identical and both models are subjected to the same training and test dataset. The next chapter describes the model structure and results obtained from both models.

CHAPTER 4 MODEL CONSTRUCTIONS AND RESULTS

4.1 Experimental Data

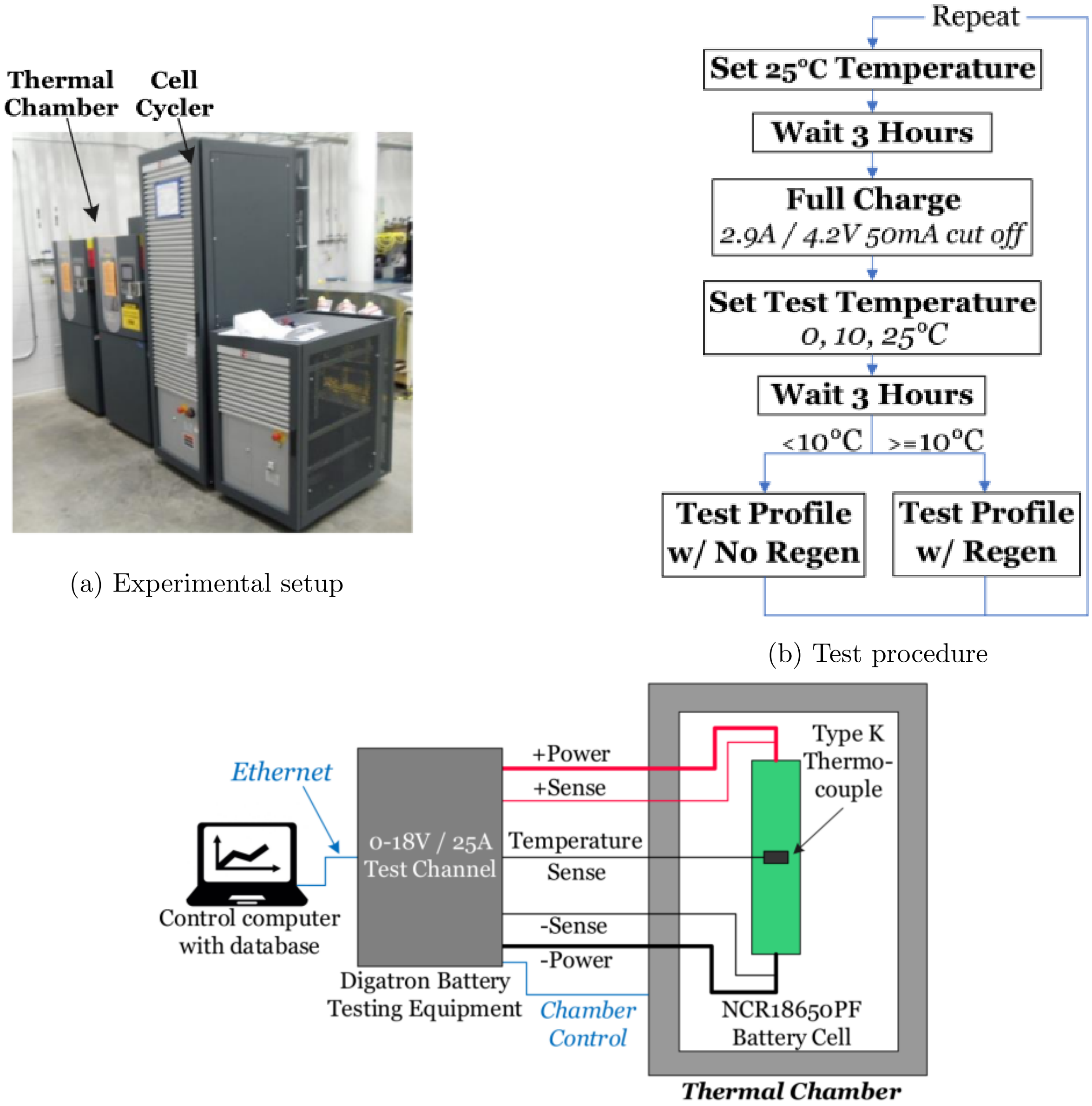
For any machine learning model, arguably the most important element is the data on which model can be trained to learn system behavior. In this study, experimental data collected to train the cell model is collected in a lab setup.

Experimental data is collected from Panasonic 18650 cell[40]. Cell chemistry is lithium nickel cobalt aluminum oxide (LiNiCoAlO2 or NCA). This battery is rated at 43 mΩ . Cell specifications are given in Table 4.1[40].

Nominal Open Circuit Voltage	3.6V
Capacity	Min. 2.75 Ah/Typ. 2.9 Ah
Min/Max Voltage	2.5V/4.2V
Mass/Energy Storage	48 gm/9.9 Wh
Minimum Charging Temperature	10°C
Cycles to 80% Capacity	500(100 % DOD, 25°C)

Table 4.1: Panasonic 18650 Cell Specifications

To generate experimental data, the battery was exposed to a range of ambient temperatures from 0°C to 25°C. The methodology for data collection is as follows: (i)Set the thermal chamber, shown in Figure 4.1a, to 25°C and wait for 3 hours so that battery internal temperature can reach 25°C. (ii) Fully charge the battery with a constant current charge rate of 2.9A, followed by a constant voltage charge at 4.2 V and terminate charging when current falls below 50 mA. (iii)Set thermal chamber temperature to desired test temperature and wait for 3 hours so that battery's internal temperature can reach test temperature. (iv)Discharge the battery using a drive cycle power profile. This methodology is summarized in Figure 4.1b.



(a) Experimental setup

(b) Test procedure

(c) Schematic of test bench and data logging system

Figure 4.1: Experiment outline summary

The drive cycle power profile is created by scaling the power needs of a vehicle to a single cell. The test vehicle considered is an electric Ford F150(not production-intent). The battery pack considered is 35kWh consisting of 3680 cells of Panasonic 18650[47,48].

4.2 Model Construction

In this work, two main categories are neural networks are constructed to predict the SOC of Li-ion battery. These 2 categories are: (i) 3 feature models: Model has 3 input features - Voltage, Current, and Battery Cell Surface Temperature. The model has 2 layers: The first layer is made up of 27 LSTM units. The second layer is made up of 1 Dense unit with “ReLU” activation. (ii) 4 feature models: Model has 4 input features - Voltage difference, Voltage, Current, and Battery Cell Surface Temperature. The extra feature of the Voltage difference measures the change in Voltage between the current reading and the last reading. Other than the extra input feature, the rest of the model construction is identical to the previous 3 feature models: The first layer with 27 LSTM units and the second layer of 1 Dense unit with “ReLU” activation.

3 feature model has a total of 3376 learn-able parameters while 4 feature model has a total of 3484 learn-able parameters. Additional learnable parameters are due to additional features. The figure 4.2 shows the construction of 3 feature and 4 feature models graphically.

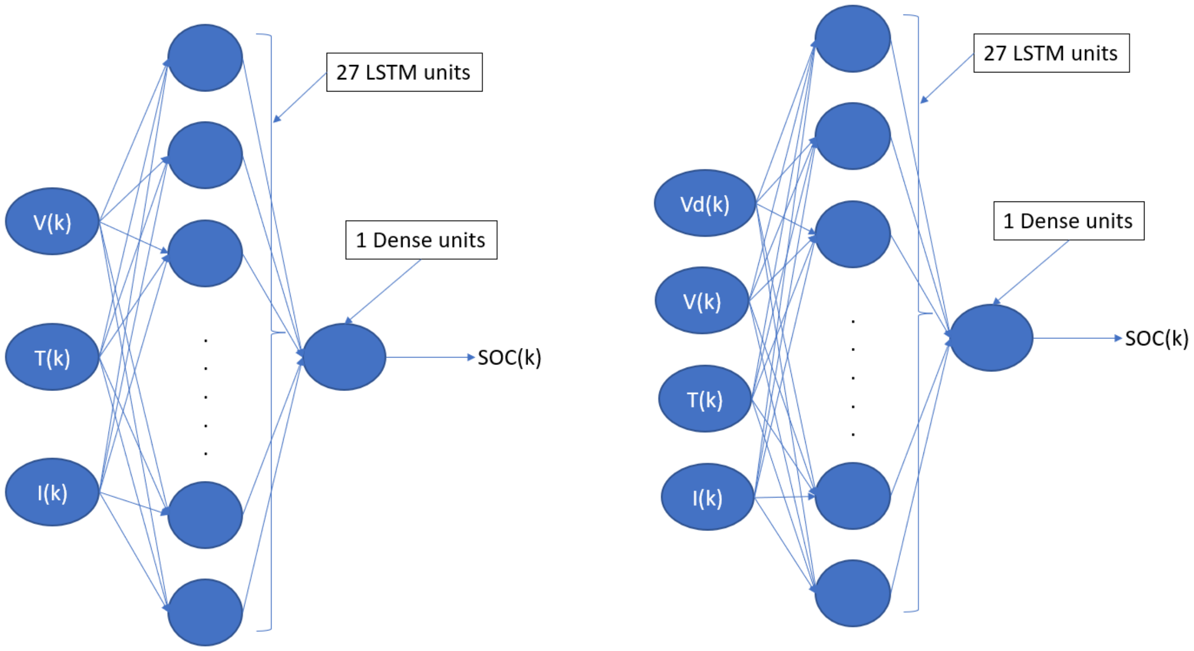


Figure 4.2: 3 feature and 4 feature model construction

4.3 Voltage difference as a feature

In this work, for 4 feature models, the Voltage difference is selected as the additional feature in addition to 3 features. The hypothesis which is being tested here is that if the model is supplied with additional insight into whether the voltage is increasing or decreasing during a short period of time, it might improve the model performance. In this work, this hypothesis will be tested by exposing 3 features and 4 feature models to the same set of test and training data, similar model architecture, and same hyper-parameter tuning and comparing their results.

4.4 Correlation between model inputs and outputs

Neural networks map the model input to output by adjusting network parameters. Data correlation is a study of understanding dependency between two variables. Spearman Correlation is often used to understand the strength of the relationship between two variables. Spearman Correlation can be applied when the data being considered may have a non-Gaussian distribution [49]. Spearman correlation coefficient ranges from 1 to -1. A correlation coefficient of 1 suggests very strong positive correlation while -1 suggests a very strong negative correlation. A coefficient of 0 indicates no correlation between those 2 variables.

In the Figure 4.3, a heatmap is plotted describing the correlation between various attributes (Voltage Difference, Voltage, Temperature, Current, and SOC) of the training data for the LA92 drive cycle at 25°C. As expected, Voltage has a very strong positive correlation (0.96) with SOC. Voltage difference has a weaker correlation with SOC (-0.0024). In this work, an attempt is made to find out whether combining Voltage difference(a feature which has a very weak direct correlation with SOC), when combined with other 3 features(Voltage, Temperature, Current) can help to improve model performance with various hyper-parameters.

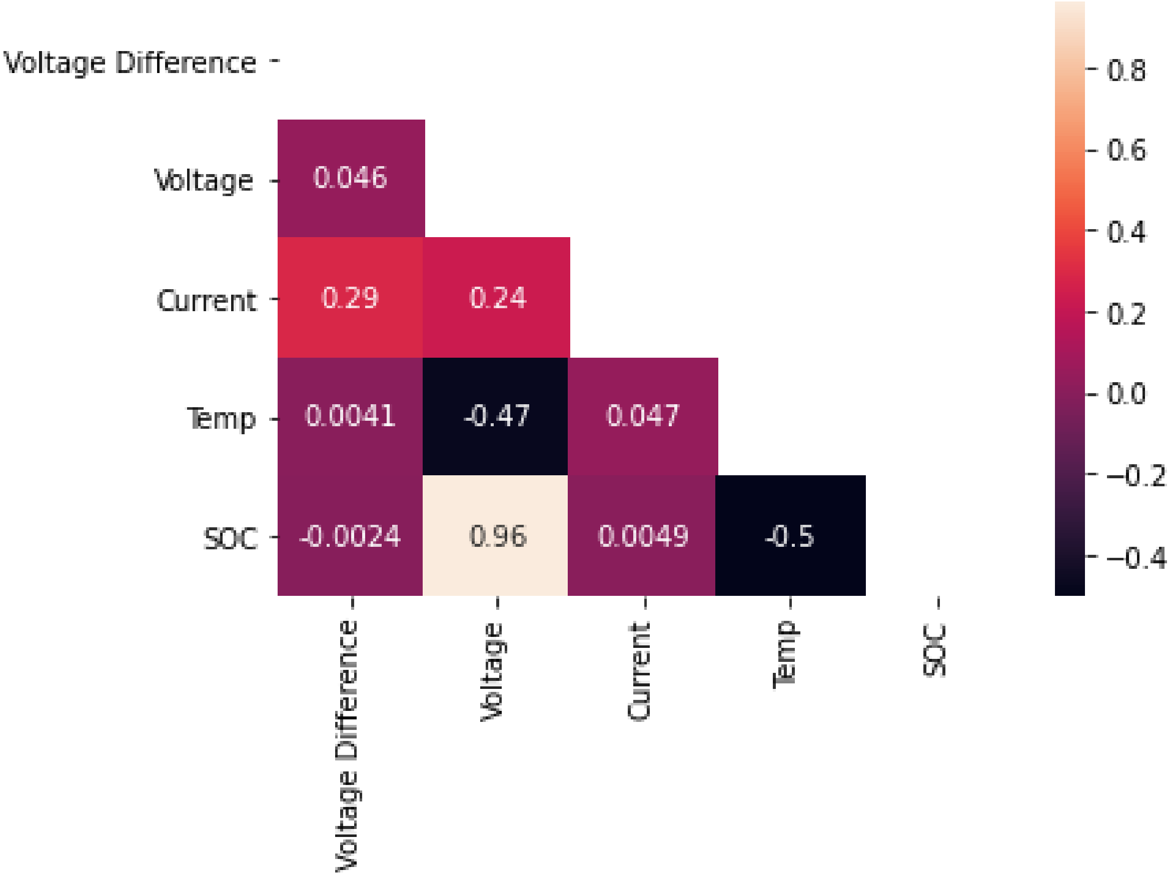


Figure 4.3: Heatmap for Correlation Coefficient

4.5 Data Preparation

LSTM-RNN model will need training data to learn model parameters that can reflect the physical behavior of the Li-ion battery. This data is provided in a matrix form with each column representing one of the features. The steps involved are below:

1. Raw data [40] is converted to MATLAB table format with size nx4. n= number of timesteps recorded. The 4 columns are Voltage, Current, Temp, SOC. SOC is calculated using the Coulomb counting method.

2. During the experiment, data is recorded at a frequency of 10 Hz. Such high-frequency data for this training. Data recorded at every second works fine. So, data is trimmed and only 1 reading out of every 10 readings is selected.

3. For model with 4 features: 1 more column is added before the Voltage for Voltage difference. This is the difference between the Voltage of the current time step and

the previous timestep. The voltage gradient at t=0 will be 0.

4. Now, matrix containing necessary training data is available: input features and output results. This data is converted to Panda DataFrames.

5. Since Voltage, Temp, Current, SOC, and Voltage difference is in different numerical ranges, they need to be scaled in either 0 to 1 or -1 to 1 depending on the feature. Scaling the data helps in reducing training efforts. Once the model is trained using this scaled data, model output(SOC) will also be scaled. It will need to be re-scaled to original values.

6. Input features(X) and outputs (Y) are separated into two data tables. In this work, python library Pandas is used for storing and managing tabular data.

7. To use this data for LSTM-RNN, it needs to be converted into a different shape such that it has columns for past time steps as well. In this model, past 400 time steps are considered for both 3 feature and 4 feature models.

8. For the LSTM training, X matrix is converted into a 3D matrix of shape [samples, timesteps, features].

   Now, this data can be used for LSTM model training. For validation, the same treatment is applied for test data.

4.6 Model Training

Chemali et. al[40] collected data as various ambient temperatures ranging from 25°C to -20°C over a period of few months. During this experimentation, the battery is expected to degrade due to many charges and discharges. To ensure that data is collected at the consistent State of Health, 25°C and 10°C data were used as they were collected in a closer time frame. Here, 25°C and 10°C represent the battery surface temperature. This is achieved by setting the battery climatic temperature set point to battery surface temperature.

For both temperatures, the following drive cycles were executed in battery cycle: Urban Dynamometer Driving Schedule (UDDS), the Highway Fuel Economy Driving

Schedule (HWFET), the Unified Driving Schedule (LA92), and the Supplemental Federal Test Procedures or US06 and NN(Neural network). NN is a custom drive profile specifically created to train the battery model and it had additional dynamics useful for training the neural network. Additionally, Cycle 1,2,3 and 4 were created by joining various parts of the standard drive cycles[50]. These cycles were chosen to expose the cell to a wide range of root mean square(RMS) current and voltage which it is expected to see during realistic usage.

For model training and validation, collected drive-cycle data was divided into two sections: Train and Test data. Tran data were used for model training and test data were used to validate the model. This approach makes sure that the model generalizes well over large data and not just learns the train data.

Train drive-cycles: LA92,UDDS,NN,Cycle 1, Cycle2, Cycle 3 and Cycle 4

Test drive-cycles: US06, HWFET

After separating the test and train drive cycles, all raw data was concated to form a single 2D matrix consisting of all train drive cycles. Then, "Data Preparation" was applied to convert the single 2D matrix to LSTM compatible 3D matrix.

4.7 Overall workflow

In this study, first, a set of base models are trained to understand their effectiveness. Later, the same base models are modified with hyperparameters L1L2, Dropout rate, and L1L2-Dropout rate combined. Updated models are trained and tested on the same train and test data respectively. Finally, these models are compared to see the performance difference between 3 feature and 4 feature models. To find the best value of the hyperparameter, a grid search is performed. Schematically, this flow is described in Figure 4.4. The number of training epochs during Grid Search is limited to 100, and during full training is limited to 500.

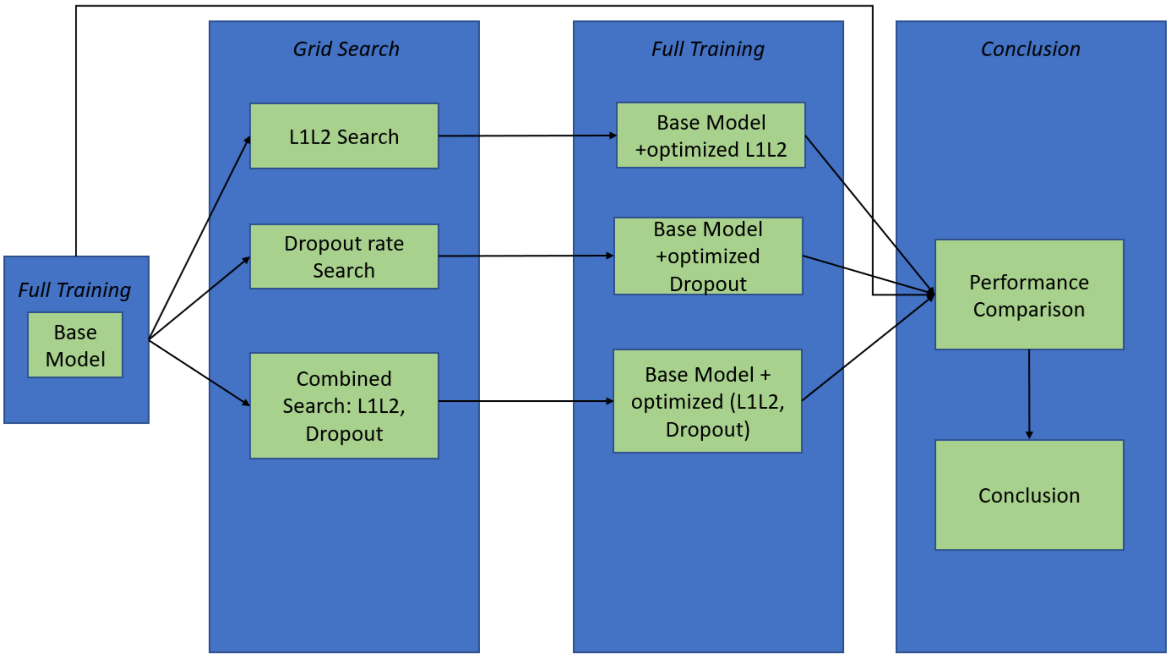
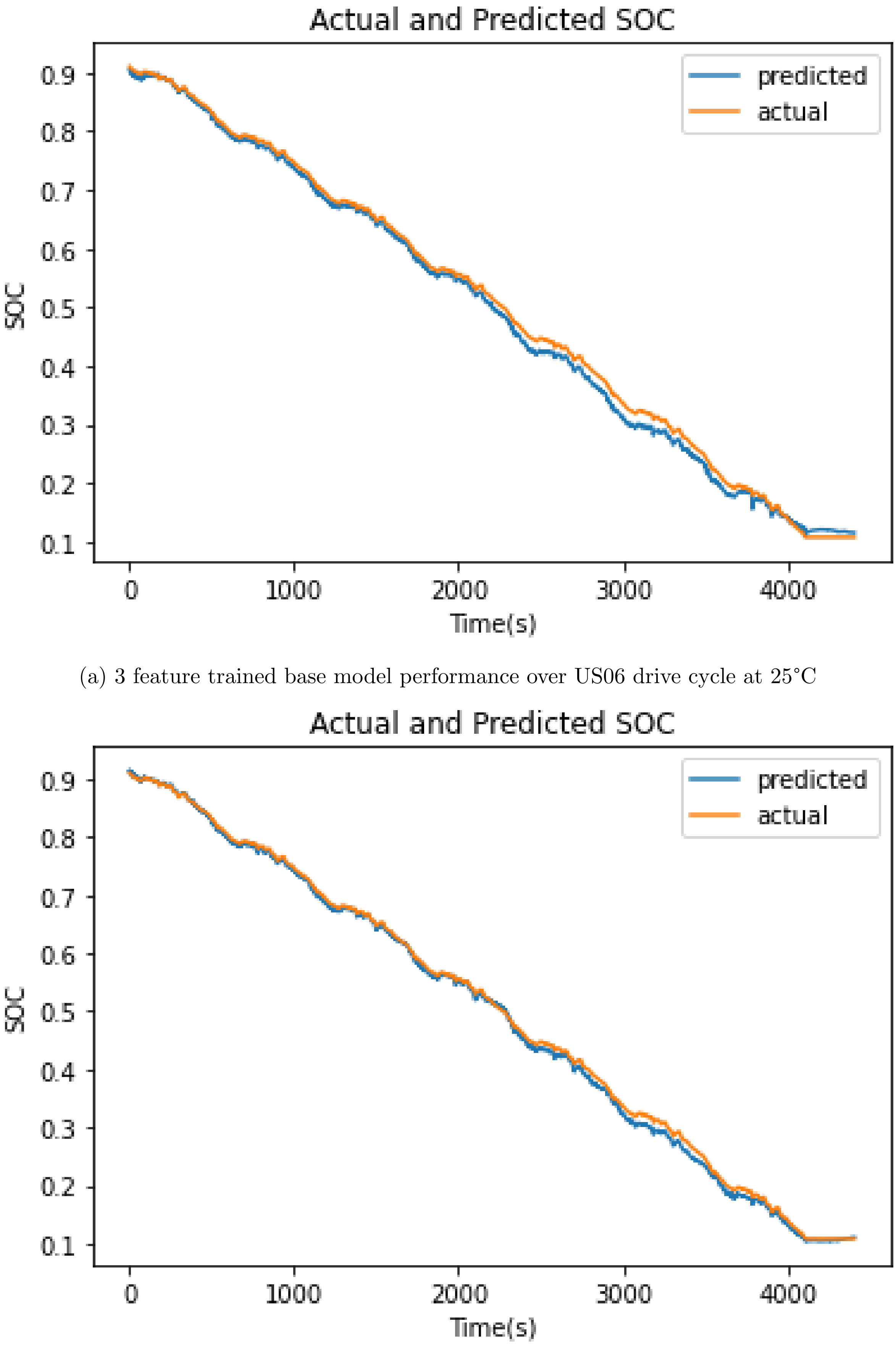

Full Training
Base Model
Grid Search
L1L2 Search
Dropout rate Search
Combined Search: L1L2, Dropout
Full Training
Base Model +optimized L1L2
Base Model +optimized Dropout
Base Model + optimized (L1L2, Dropout)
Conclusion
Performance Comparison
Conclusion

Figure 4.4: Overall workflow

4.7.1 Full training: Base Model

For base model training, 3 feature models and 4 feature models were constructed as outlined in section 4.2. Both models were trained for 500 epochs with Adam optimizer. Since training a neural network is a stochastic process, each type of model was trained 10 times and best results were considered for further review.

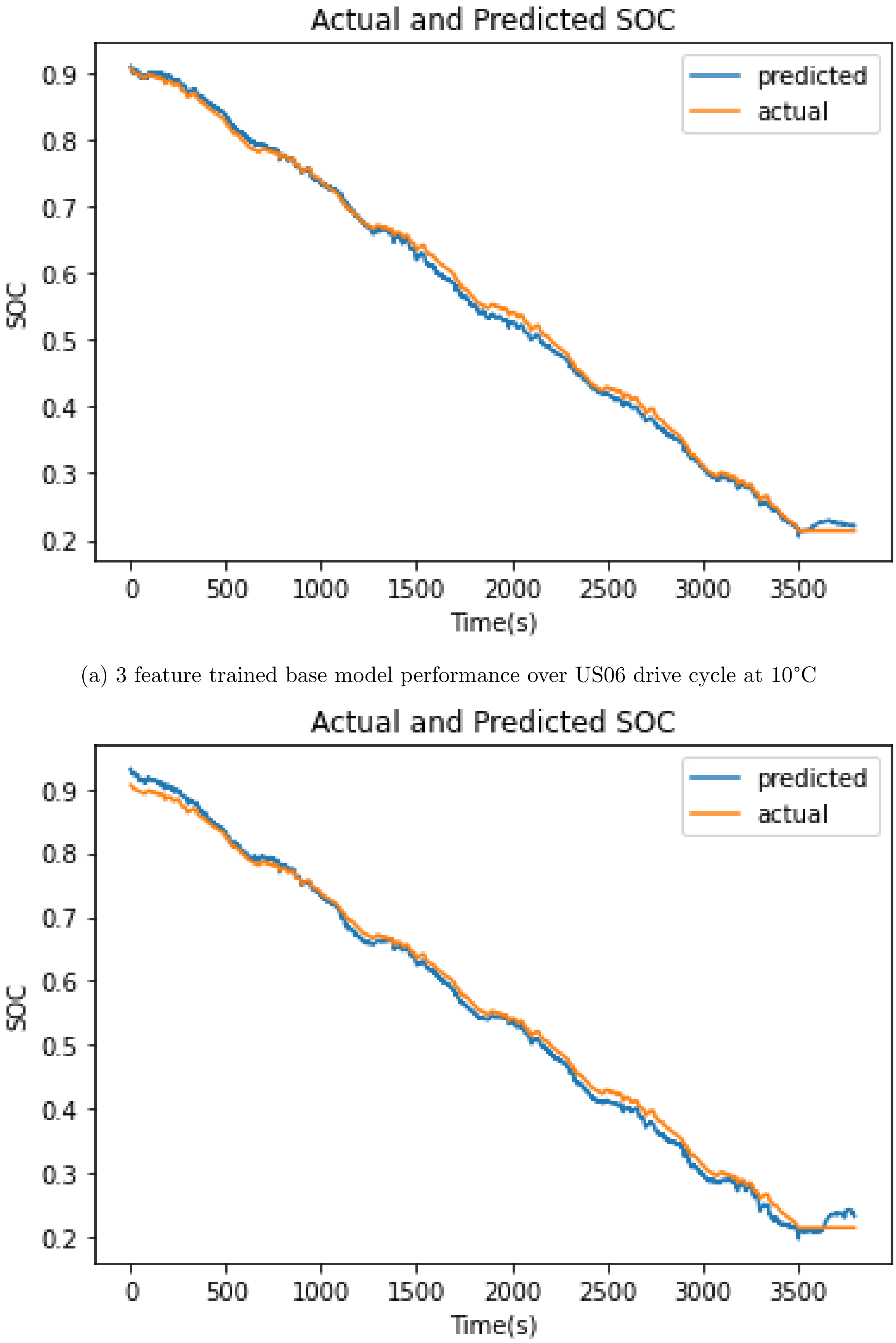
Figure 4.5 and figure 4.6 represent the performance of trained models at 25°C and at 10°C for the US06 test drive cycle respectively by plotting actual and predicted SOC. Figure 4.7 and figure 4.8 represent the percentage error during the same drive cycle at 25°C and 10°C respectively. Towards the end, the percentage error increases since the actual SOC decreases, and even a small absolute error in SOC gets divided by a small quantity and magnifies the percentage error. From these plots, it becomes clear that even with relatively modest training of 500 epochs, the model is able to replicate battery behavior.



(a) 3 feature trained base model performance over US06 drive cycle at 25°C

(b) 4 feature trained base model performance over US06 drive cycle at 25°C

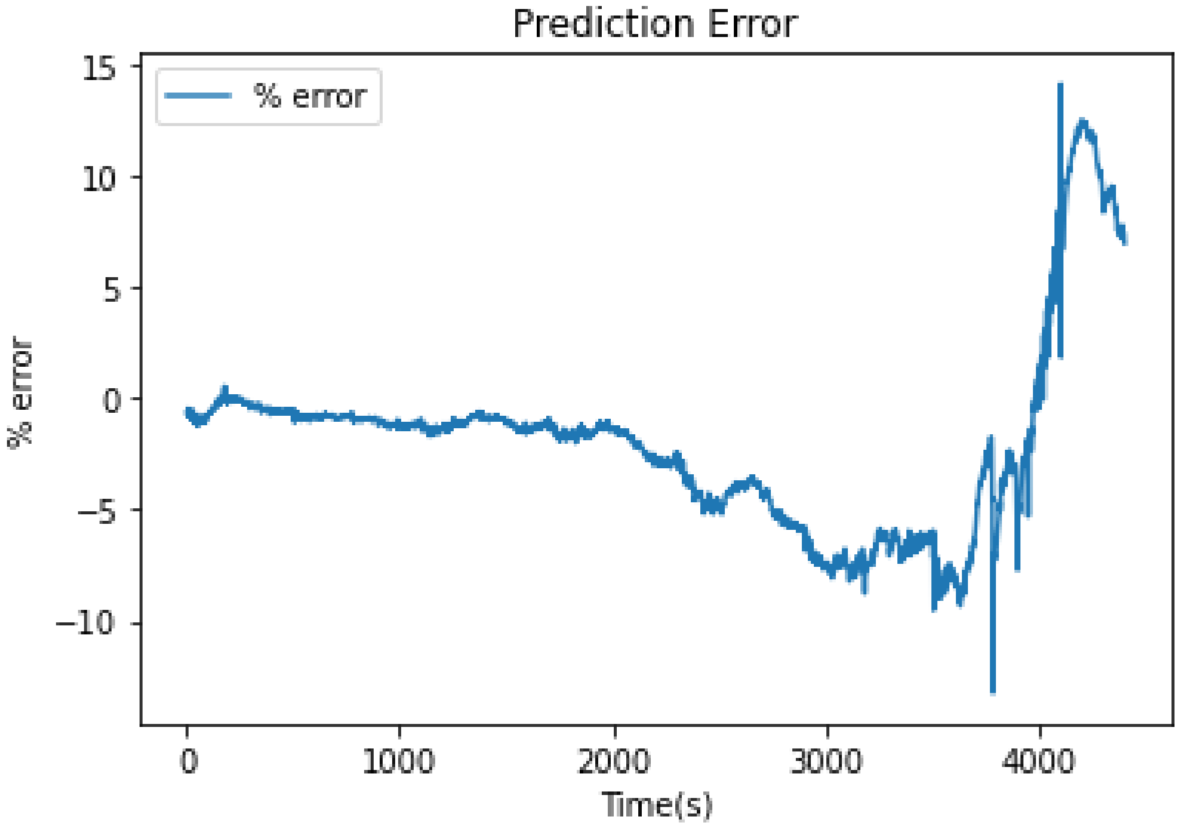
Figure 4.5: Trained base models at 25°C



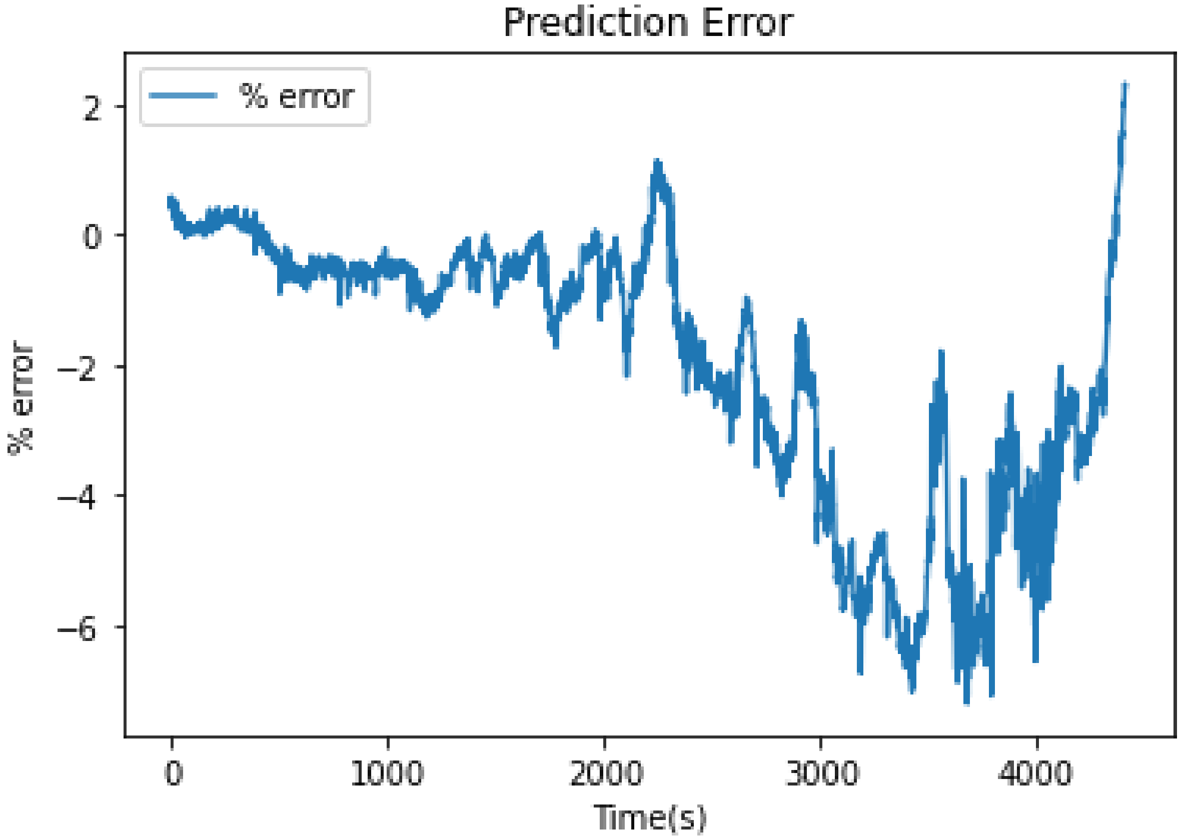
(a) 3 feature trained base model performance over US06 drive cycle at 10°C

(b) 4 feature trained base model performance over US06 drive cycle at 10°C

Figure 4.6: Trained base models at 10°C

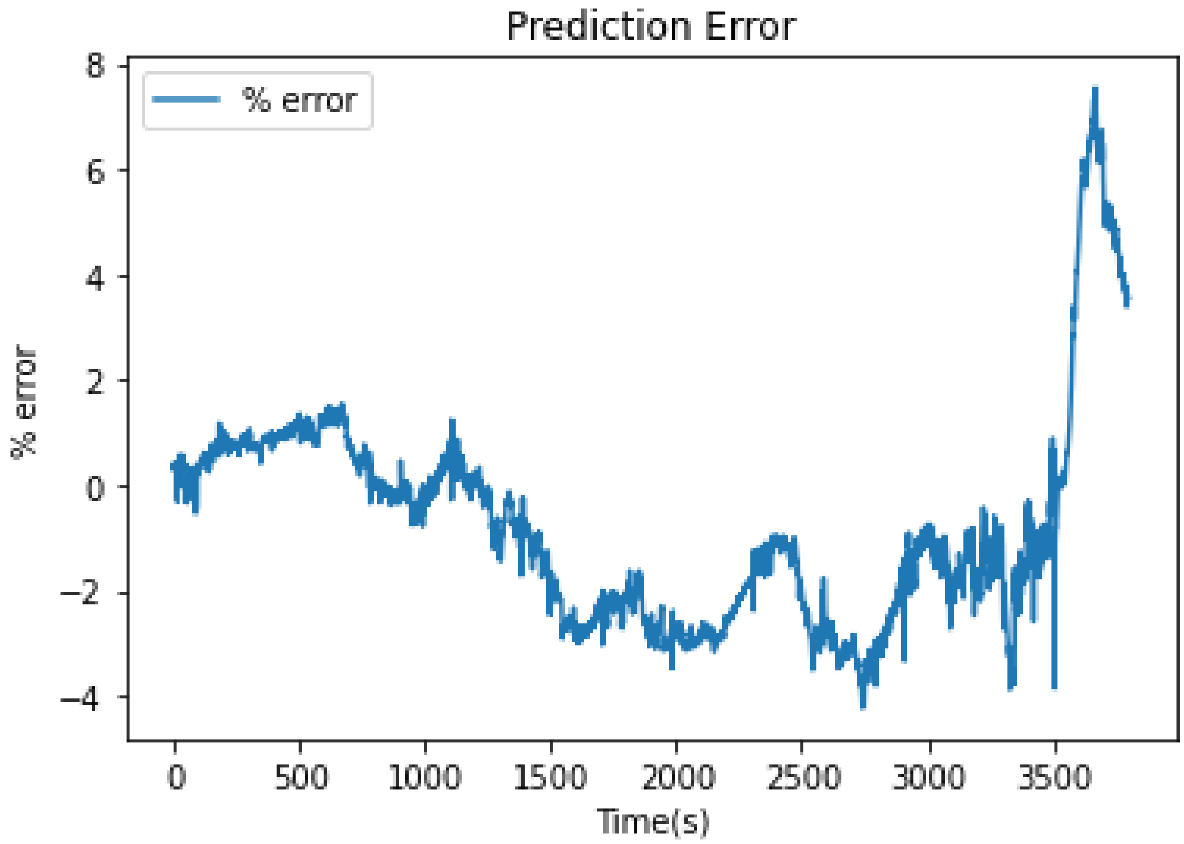


(a) 3 feature trained base model percentage error over US06 drive cycle at 25°C

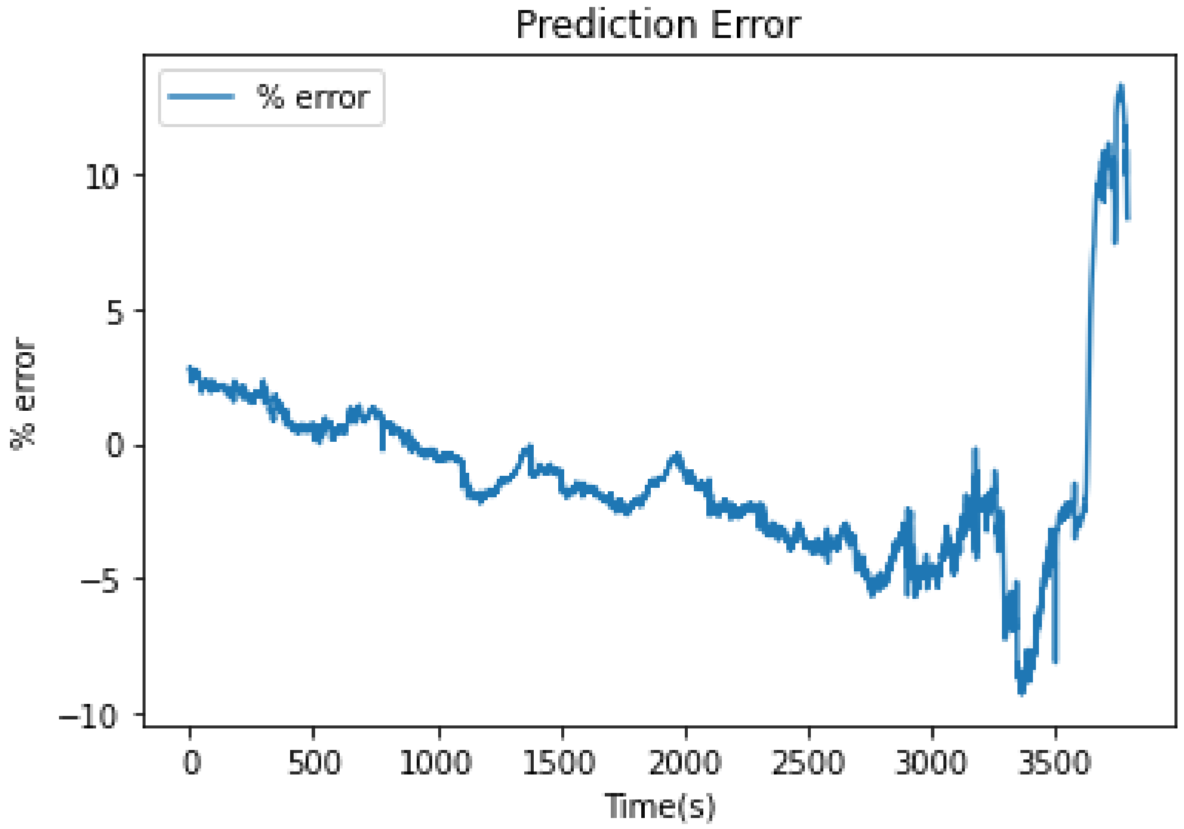


(b) 4 feature trained base model percentage error over US06 drive cycle at 25°C

Figure 4.7: Trained base models percentage error at 25°C



(a) 3 feature trained base model percentage error over US06 drive cycle at 10°C



(b) 4 feature trained base model percentage error over US06 drive cycle at 10°C

Figure 4.8: Trained base models percentage error at 10°C

Figure 4.9 represents the comparison of 3 feature and 4 feature trained models over all test drive cycles. It appears that the 4 feature model performs better(less error) for 25°C test drive cycles.

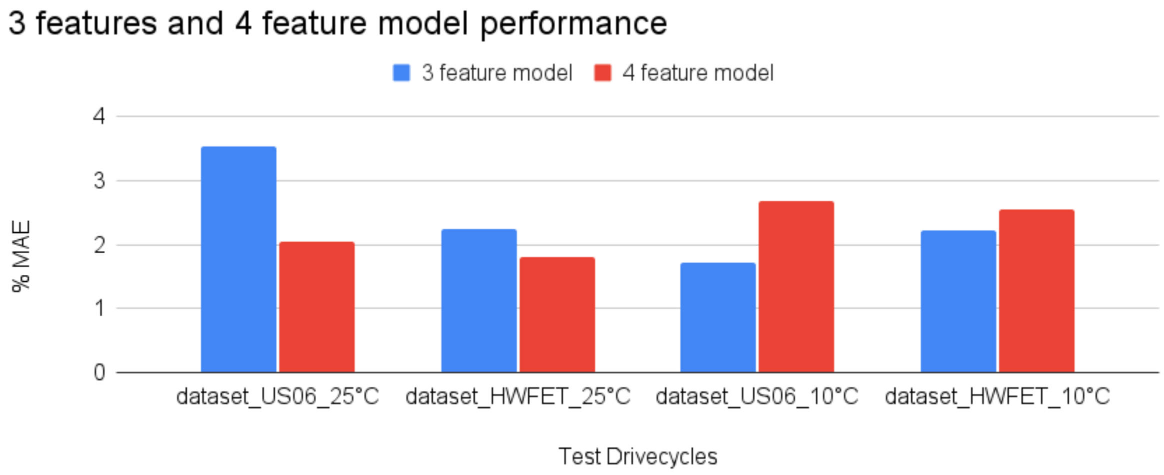


Figure 4.9: Base Models: Comparison of 3 feature and 4 feature model

4.7.2 Hyper-parameter tuning

Hyperparameter is defined as a parameter that is set before the learning process begins[51]. These parameters are tune-able and can affect model performance. Some common examples of hyper-parameters are Learning rate, Dropout rate, Weight Regularization, etc. In this study, Adam optimizer is used and since Adam optimizer automatically selects a learning rate, hence focus is mainly on finding the best Dropout rate and Weight Regularization.

Weight Regularization and L1L2 hyper-parameter search

When a neural network is trained, its weights are learned using a stochastic gradient descent process and training dataset. The longer a network is trained, network weights tend to become specialized weights which can very well fit the training data but fail to generalize. Weights grow to handle the specifics of training data. Large network weights make the network unstable as a minor variation or statistical noise on the expected inputs will result in large differences in the output. Weight regularization is a technique to prevent weights from getting big. In this technique, the cost function is modified such that large weights are penalized. In this study, L1L2 regularization is used to implement weight regularization. The modified cost function is given by :

$$Loss = Error(y, \widetilde{y}) + \lambda_1 \sum_{i=1}^{N} |w| + \lambda_2 \sum_{i=1}^{N} |w^2|$$

In this study, a grid search is performed with L1 values and L2 values from [0.0,-0.0001,0.001,0.01,0.1]. Total 25 models are generated. Each model is trained for 100 epochs with one training cycle (LA92) and the model with least loss value at end of training is considered as the model with best hyperparameters. Since the optimization process is stochastic in nature, this exercise was repeated 3 times. For 3 feature model, best value of L1L2 hyper-parameter is L1=0.0, L2=0.01 and for 4 feature model, best value of L1L2 hyper-parameter is L1=0.0, L2=0.1.

Dropout and Dropout rate hyper-parameter search

Dropout is another method to reduce overfitting to the training data. LSTM networks tend to overfit, and dropout is often used as an effective strategy to counteract those effects. Dropout works by dropping out inputs to a layer in a probabilistic manner. Thus, it produces effect of simulating many networks with different structure. This results in making network more robust to the inputs [52].

In this study, dropout is applied between LSTM layer and Dense layer. A series of dropout values are explored, [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]. Total 11 models are generated. Each model is trained for 100 epochs with one training cycle(LA92) and the model with the least loss value at end of training is considered as the model with best hyperparameters. Since the optimization process is stochastic in nature, this exercise was repeated 3 times. For both 3 feature and 4 feature model, the best value of the dropout rate is found to be 0.1.

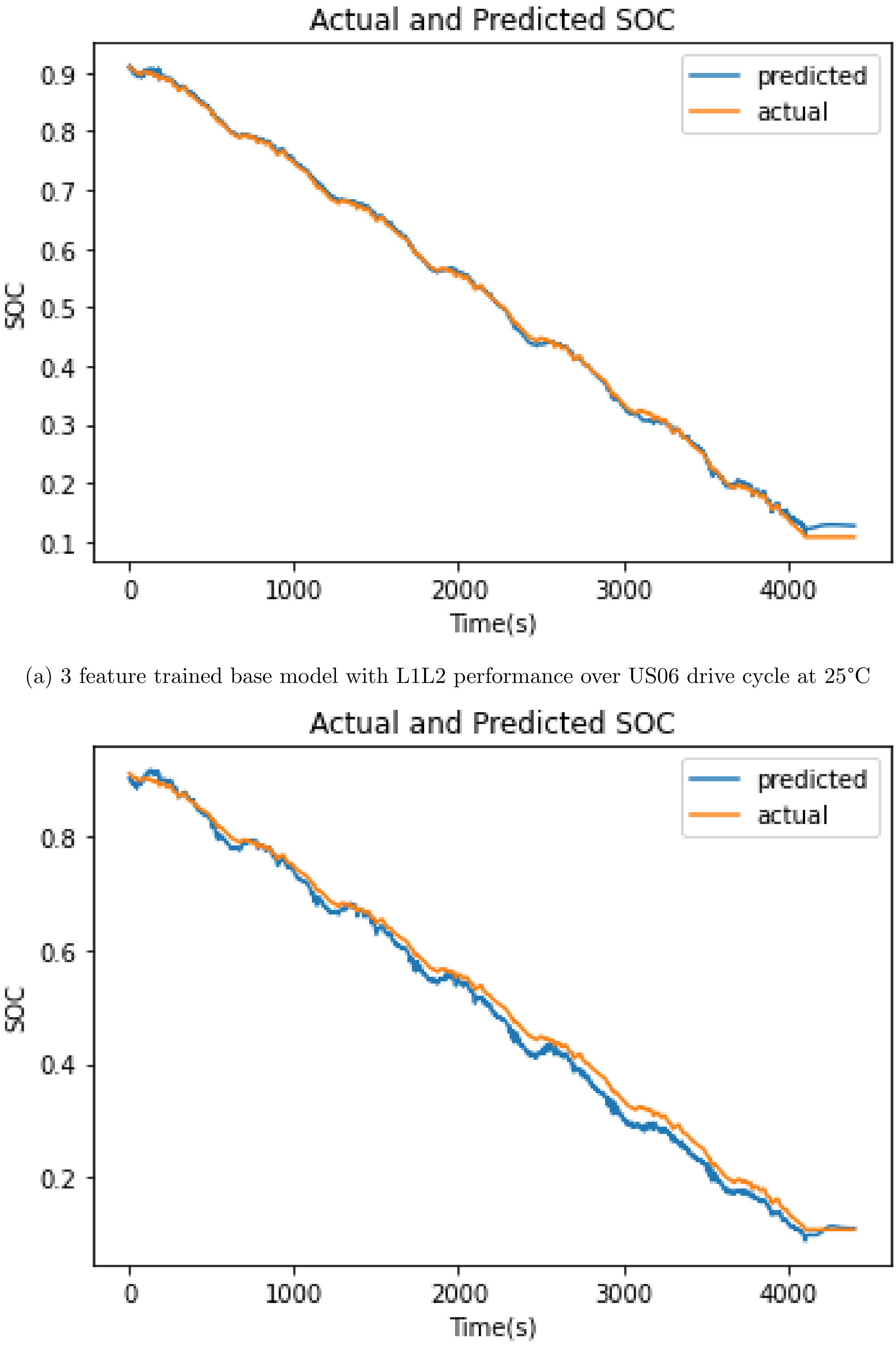
Combined Regularization: L1L2 and Dropout

In this approach, both types of hyper-parameters is applied on the base model. The range of L1L2 and Dropout rate is kept the same as in previous explorations in section 4.7.2 and 4.7.2. Total 275 models are generated and trained for 100 epochs using one training cycle(LA92) and model with the least loss value is considered as the one with best set of hyper-parameters. Since the optimization process is stochastic in nature, this exercise was repeated 3 times. Best value of hyper-parameters for 3 features was found to be L1=0.0,L2=0.001,dropout rate=0.0 and for 4 feature it was found to be L1=0.0,L2=0.01,dropout rate=0.0.

4.7.3 Full training: Base Models with L1L2 regularization

For the base model with L1L2 hyper-parameter training, the cost function was updated to incorporate L1 and L2 values. Best values of L1 and L2 for 3 feature and 4 feature model were previously searched through a Grid search process. For 3 feature model, best values of L1L2 hyper-parameter are L1=0.0,L2=0.01. For 4 feature model, best value of L1L2 hyper-parameter are L1=0.0, L2=0.1. Both models were trained for 500 epochs with Adam optimizer. Training was repeated 10 times to account for stochastic nature of training.

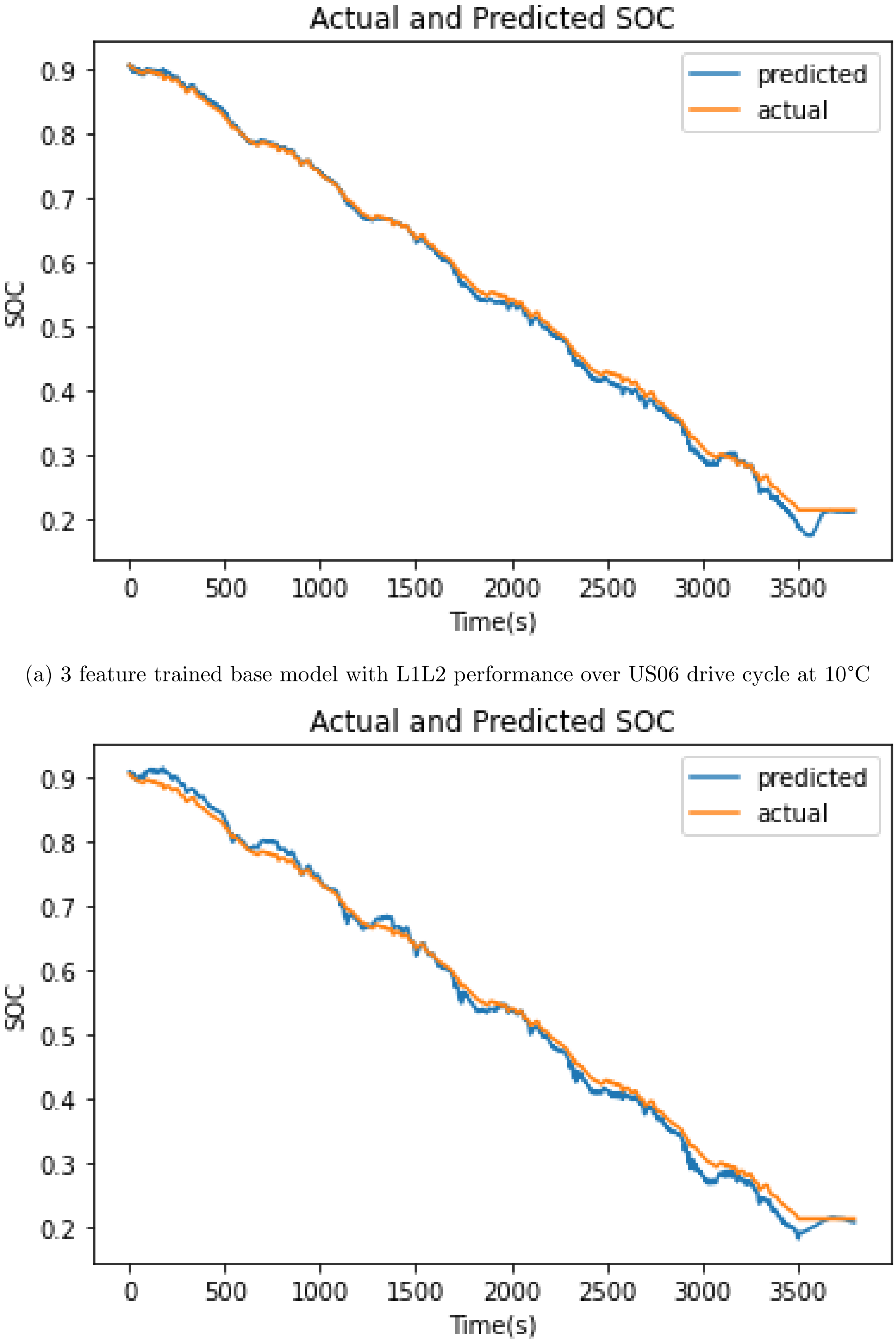
Figure 4.10 and figure 4.11 represents the performance of trained models at 25°C and at 10°C for the US06 test drive cycle respectively by plotting actual and predicted SOC. Figure 4.12 and figure 4.14 represent the percentage error during the same drive cycle at 25°C and 10°C respectively.



(a) 3 feature trained base model with L1L2 performance over US06 drive cycle at 25°C

(b) 4 feature trained base model with L1L2 performance over US06 drive cycle at 25°C

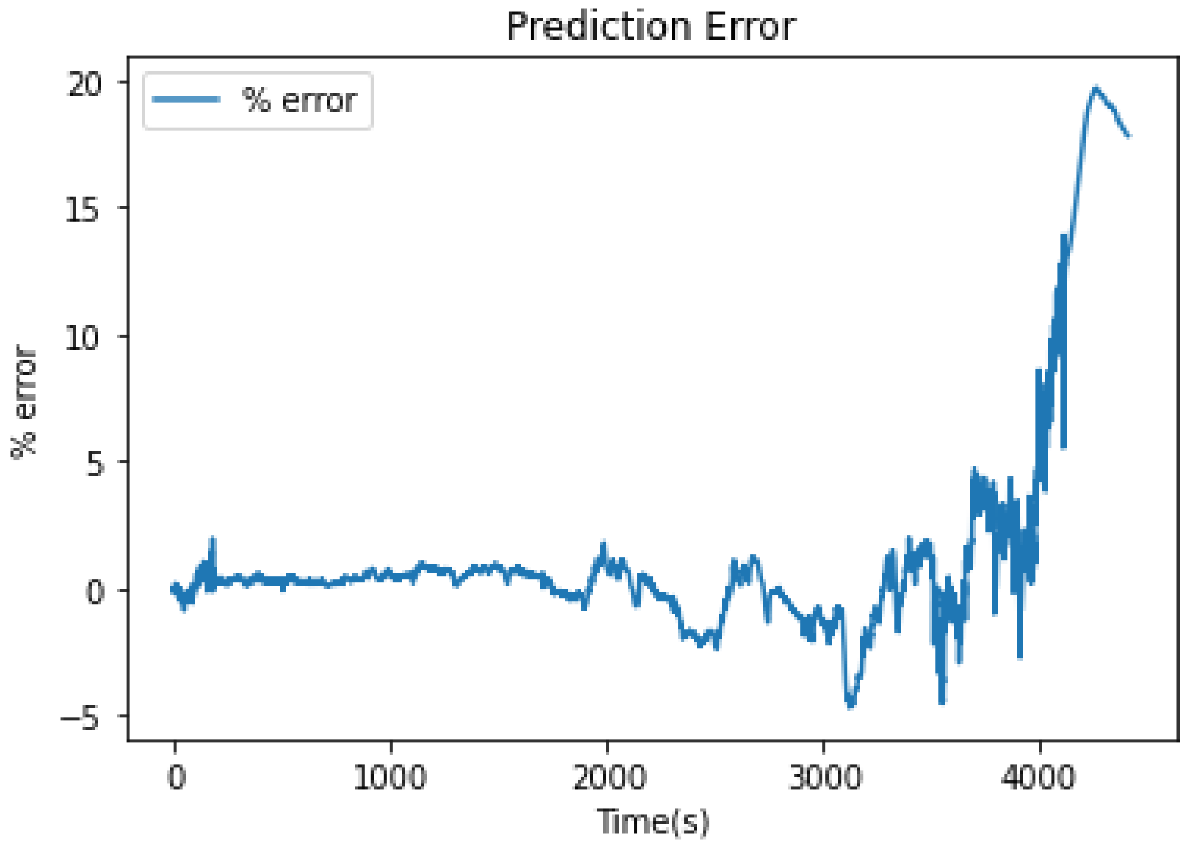
Figure 4.10: Trained base models with L1L2 at 25°C



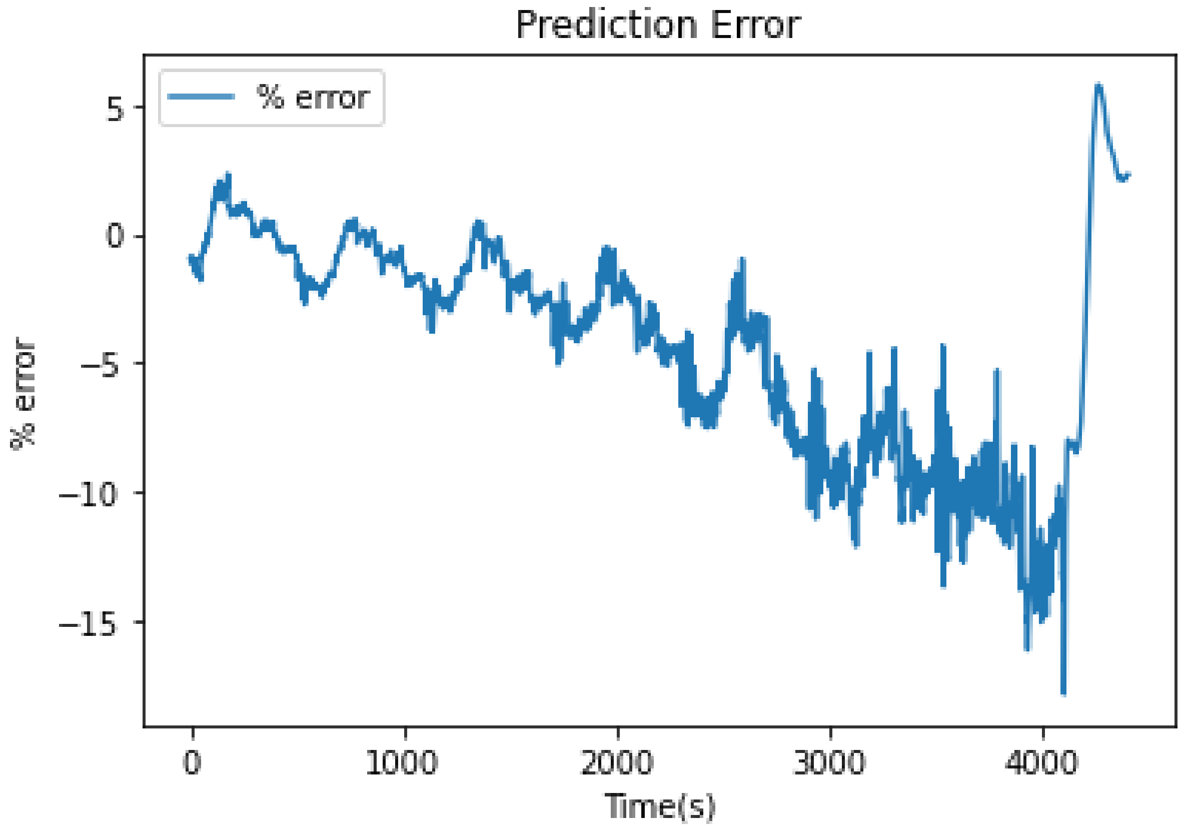
(a) 3 feature trained base model with L1L2 performance over US06 drive cycle at 10°C

(b) 4 feature trained base model with L1L2 performance over US06 drive cycle at 10°C

Figure 4.11: Trained base models with L1L2 at 10°C

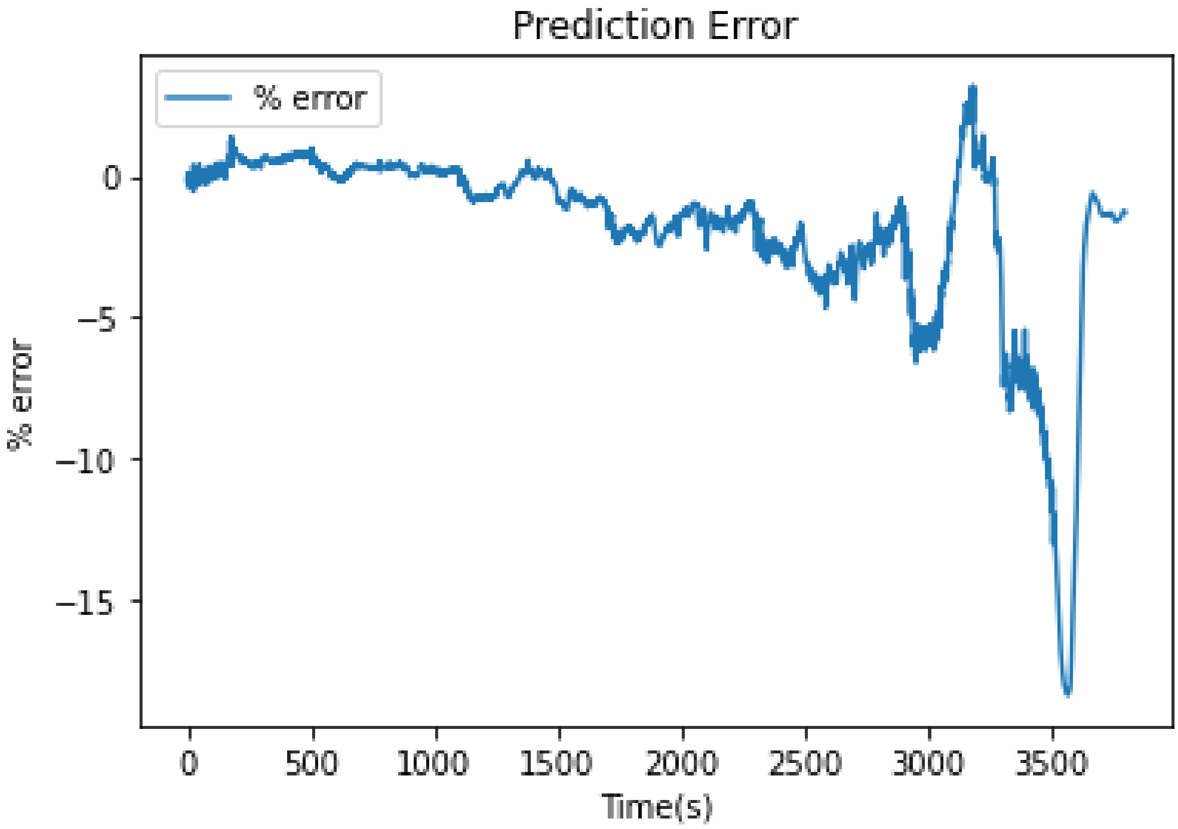


(a) 3 feature trained base model with L1L2 percentage error over US06 drive cycle at 25°C

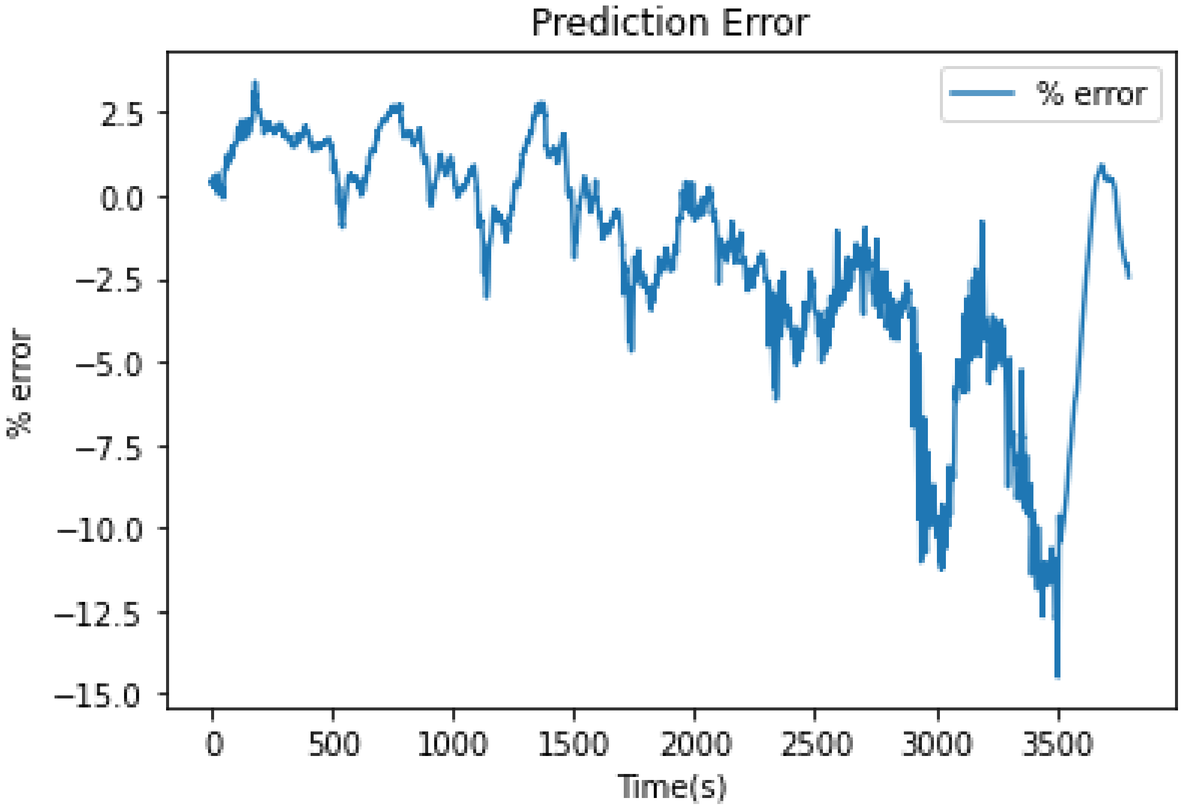


(b) 4 feature trained base model with L1L2 percentage error over US06 drive cycle at 25°C

Figure 4.12: Trained base models with L1L2 percentage error at 25°C



(a) 3 feature trained base model with L1L2 percentage error over US06 drive cycle at 10°C



(b) 4 feature trained base model with L1L2 percentage error over US06 drive cycle at 10°C

Figure 4.13: Trained base models with L1L2 percentage error at 10°C

The figure 4.14 represents the comparison of 3 feature and 4 feature trained models over all test drive cycles. It becomes clear that 3 feature models performed better for all drive cycles. An additional feature in the 4 feature model did not help to improve the performance.

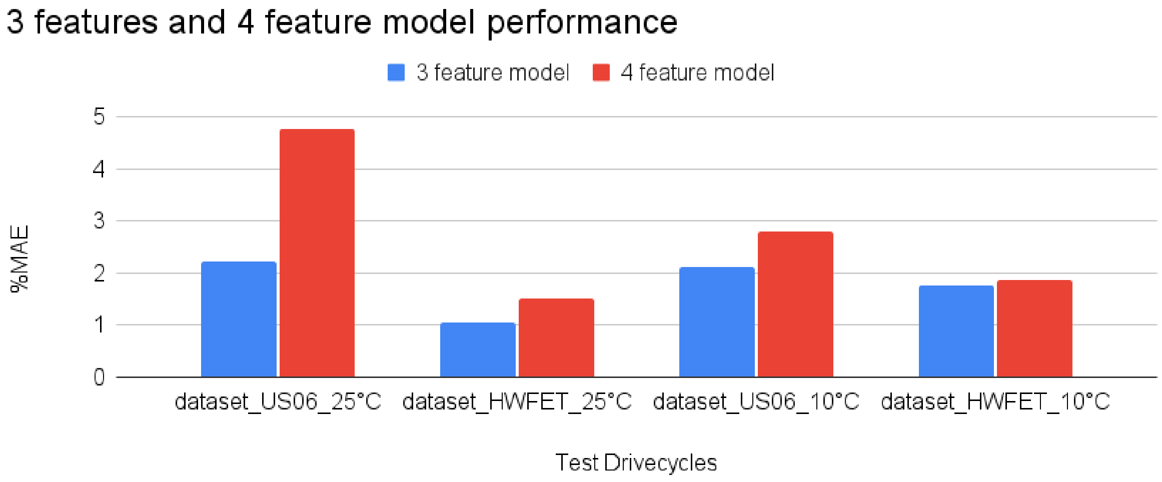
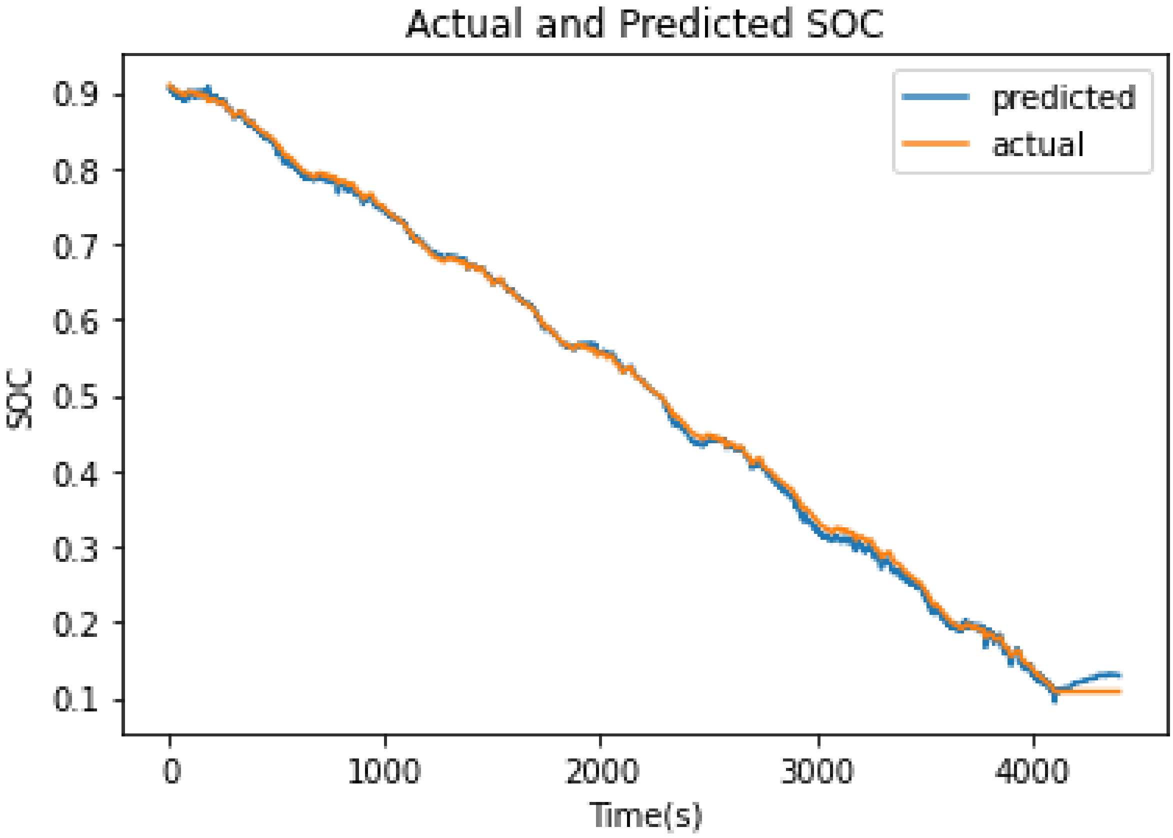


Figure 4.14: Base Models with L1L2: Comparison of 3 feature and 4 feature model

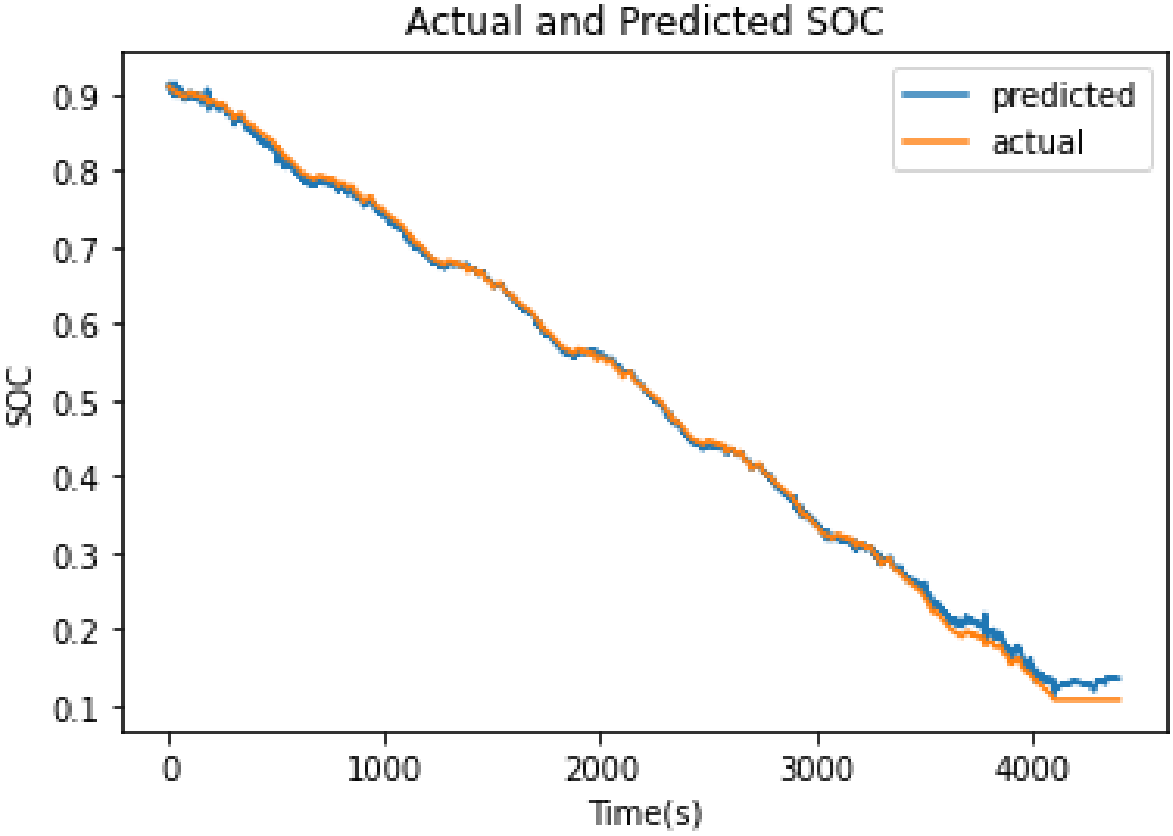
4.7.4 Full training: Base models with dropout

In this training, base model construction was updated with the dropout layer. The Grid search process returned the best values of dropout as 0.1 for both 3 feature and 4 feature models. Both 3 features and 4 features updated with dropout layer were trained for 500 epochs with Adam optimizer. Models were trained 10 times to account for stochastic nature of training.

Figure 4.15 and figure 4.16 represents the performance of trained models at 25°C and at 10°C for the US06 test drive cycle respectively by plotting actual and predicted SOC. Figure 4.17 and figure 4.18 represent the percentage error during the same drive cycle at 25°C and 10°C respectively.

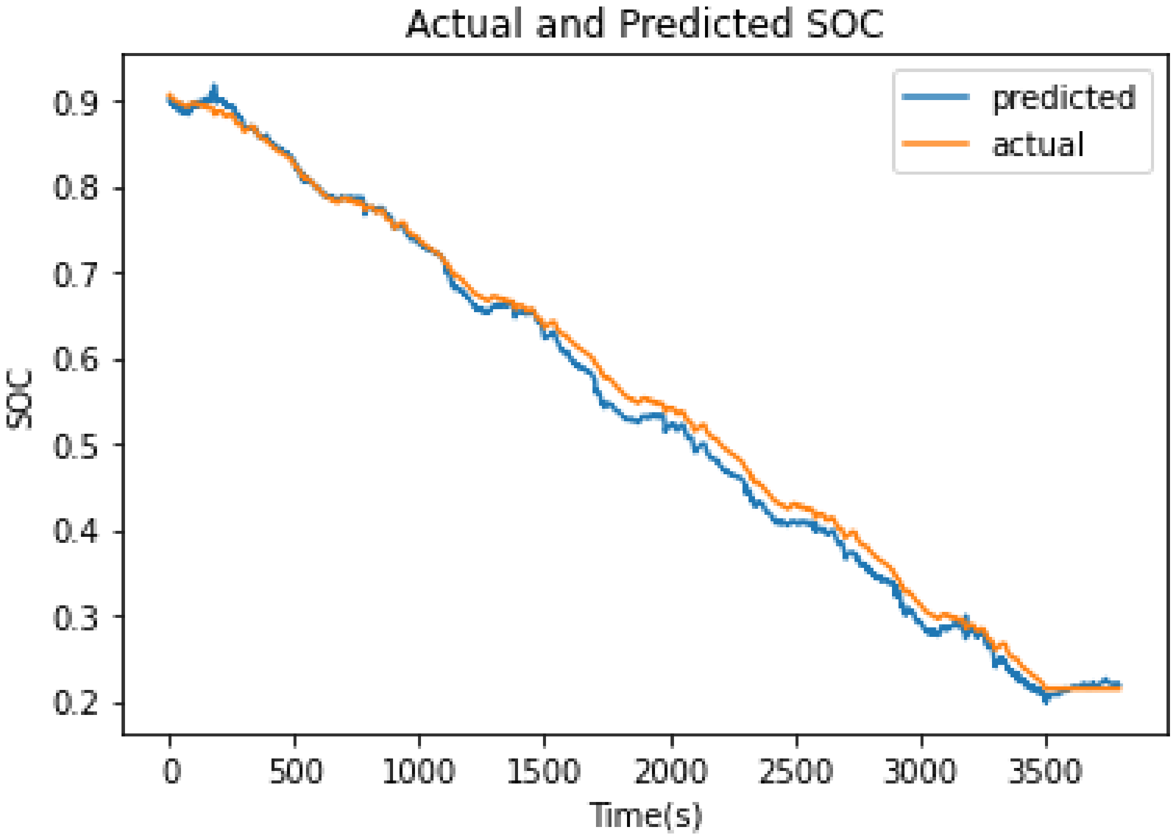


(a) 3 feature trained base model with dropout performance over US06 drive cycle at 25°C

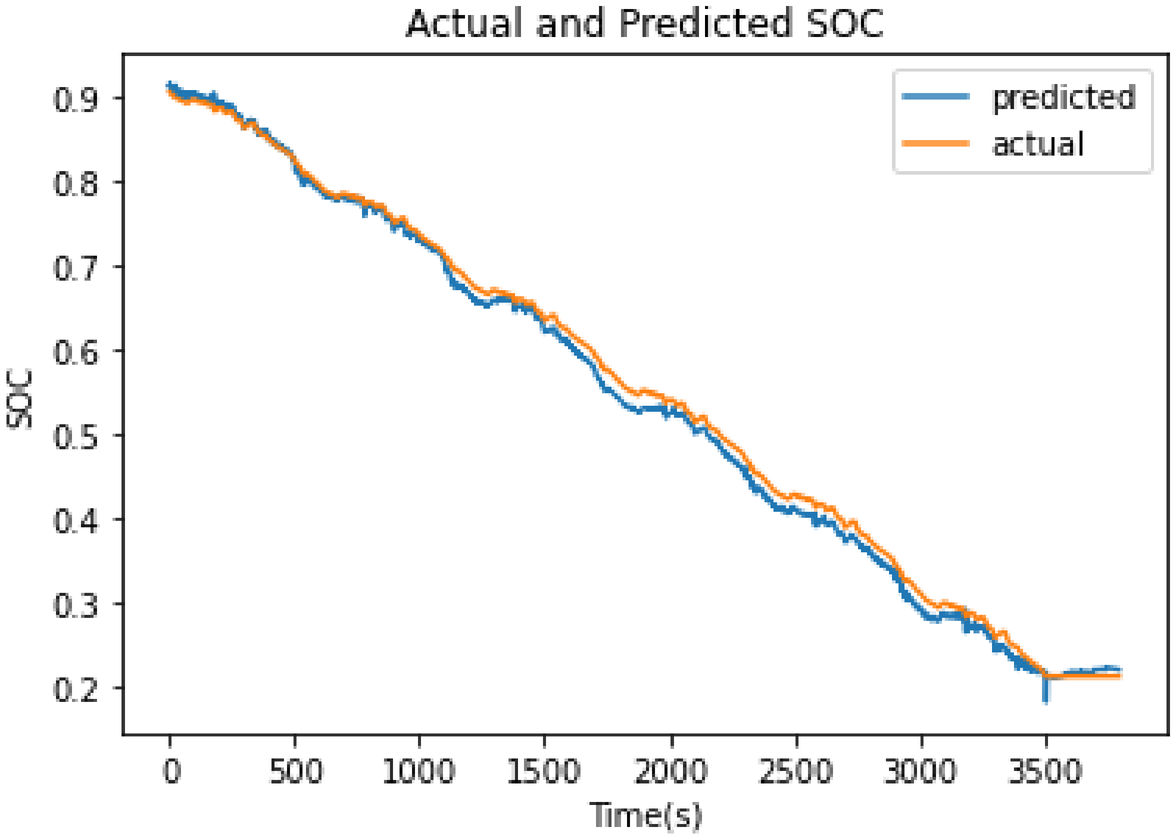


(b) 4 feature trained base model with dropout performance over US06 drive cycle at 25°C

Figure 4.15: Trained base models with dropout at 25°C

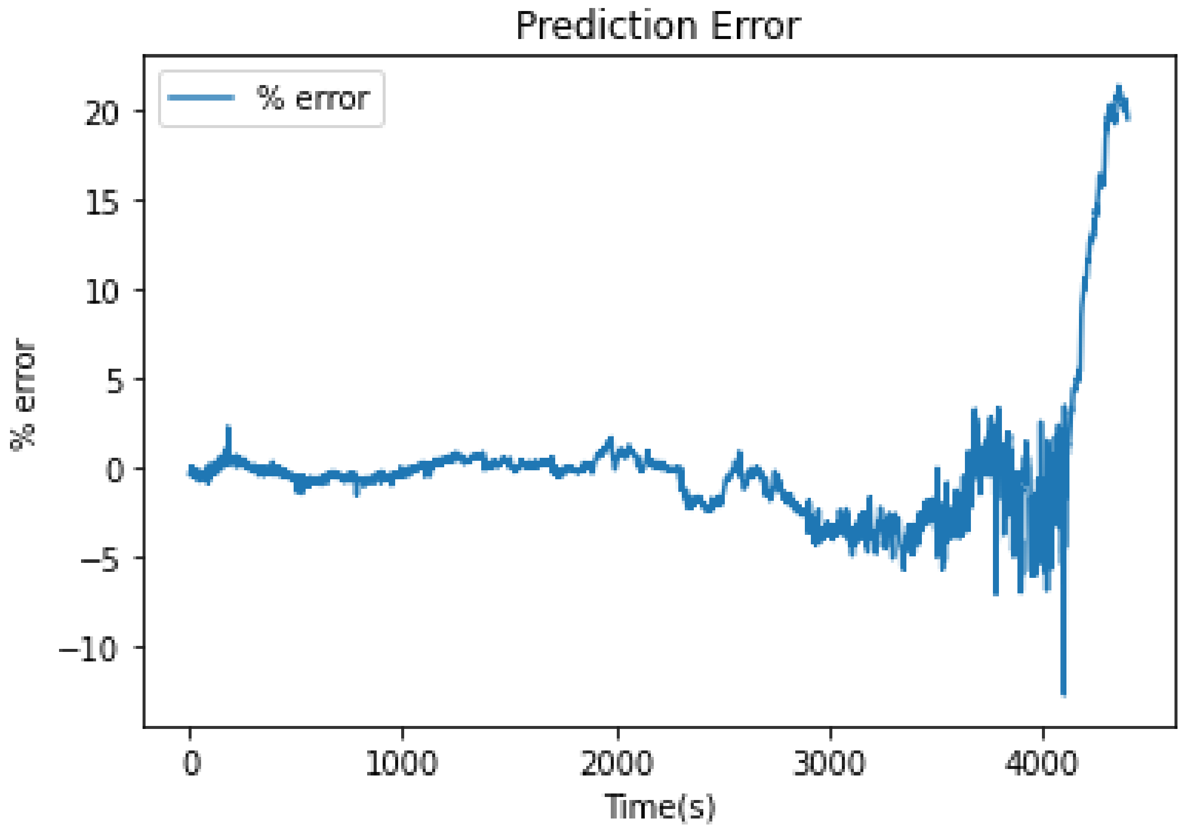


(a) 3 feature trained base model with dropout performance over US06 drive cycle at 10°C

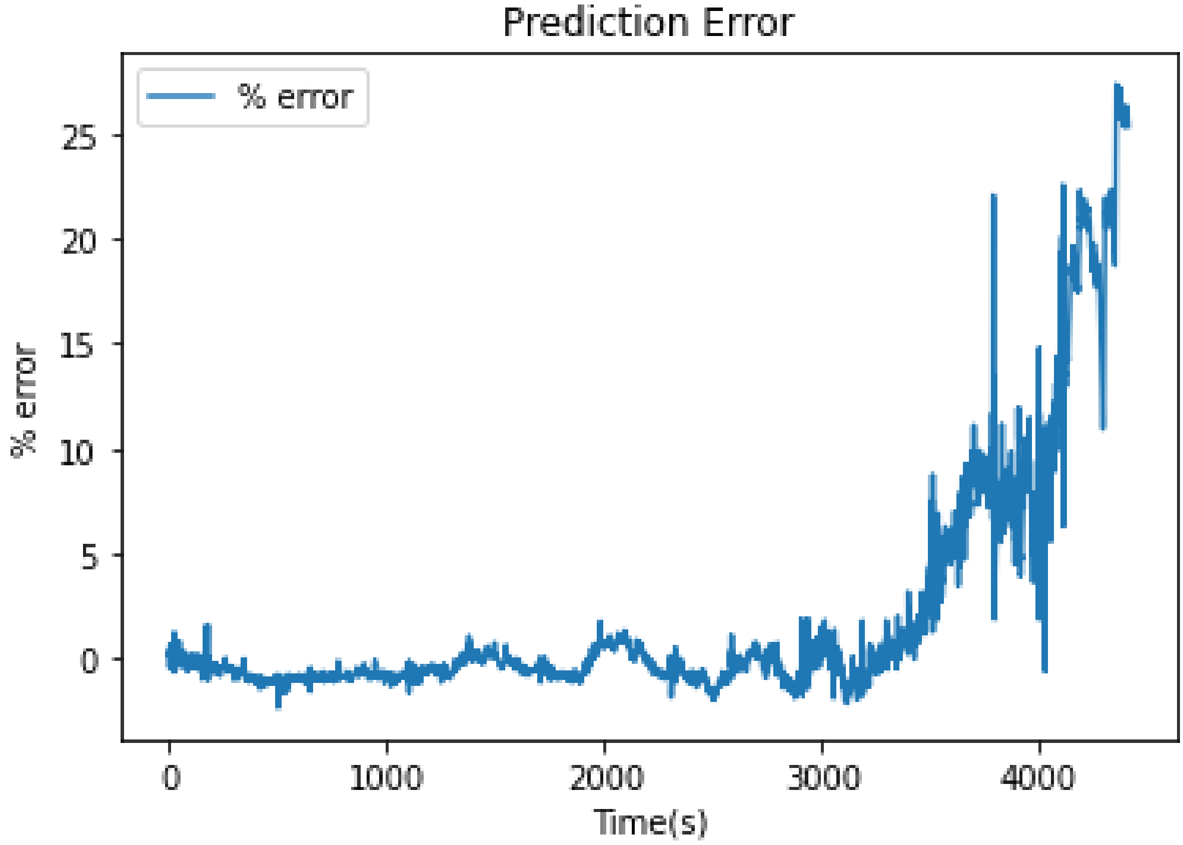


(b) 4 feature trained base model with dropout performance over US06 drive cycle at 10°C

Figure 4.16: Trained base models with dropout at 10°C

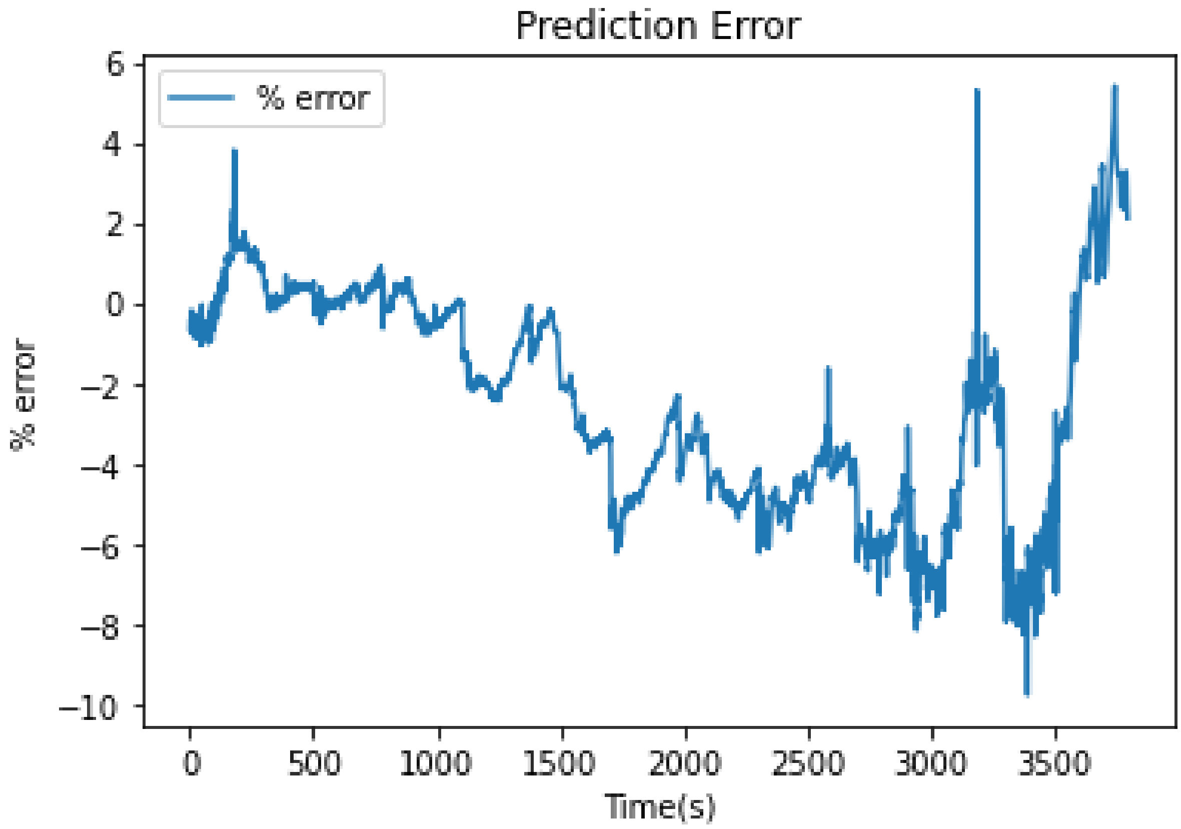


(a) 3 feature trained base model with dropout percentage error over US06 drive cycle at 25°C

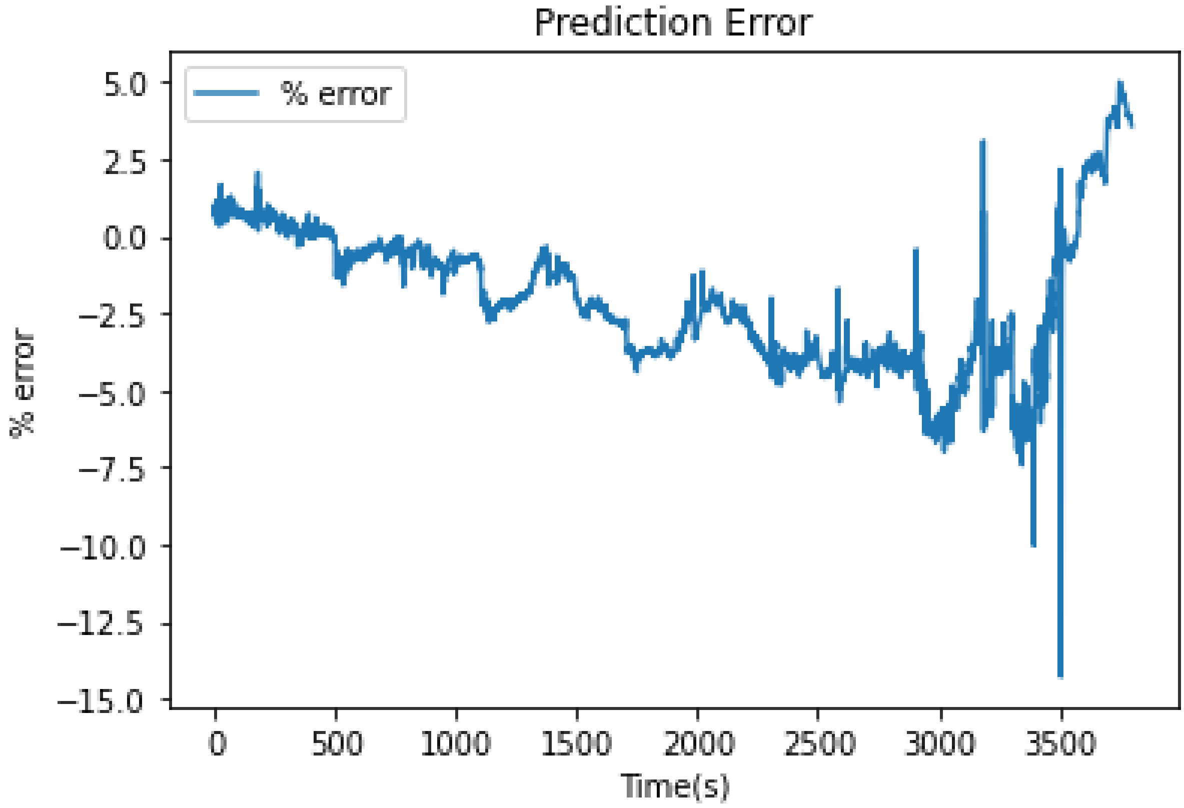


(b) 4 feature trained base model with dropout percentage error over US06 drive cycle at 25°C

Figure 4.17: Trained base models with dropout percentage error at 25°C



(a) 3 feature trained base model with dropout percentage error over US06 drive cycle at 10°C



(b) 4 feature trained base model with dropout percentage error over US06 drive cycle at 10°C

Figure 4.18: Trained base models with dropout percentage error at 10°C

Figure 4.19 represents the comparison of 3 feature and 4 feature trained models over all test drive cycles. In this training, mixed results were obtained. 3 feature model performed better for 2 test drive cycles while 4 feature model performed better for remaining 1 test drive cycles. Both models performed equal for remaining 1 cycle. However, improvement due to 4 feature model was very marginal for that drive cycle.

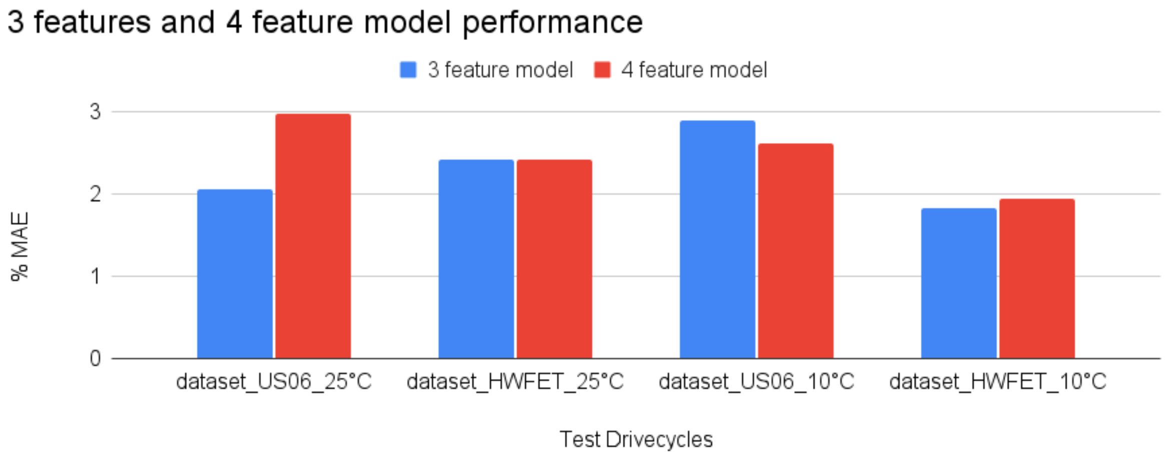
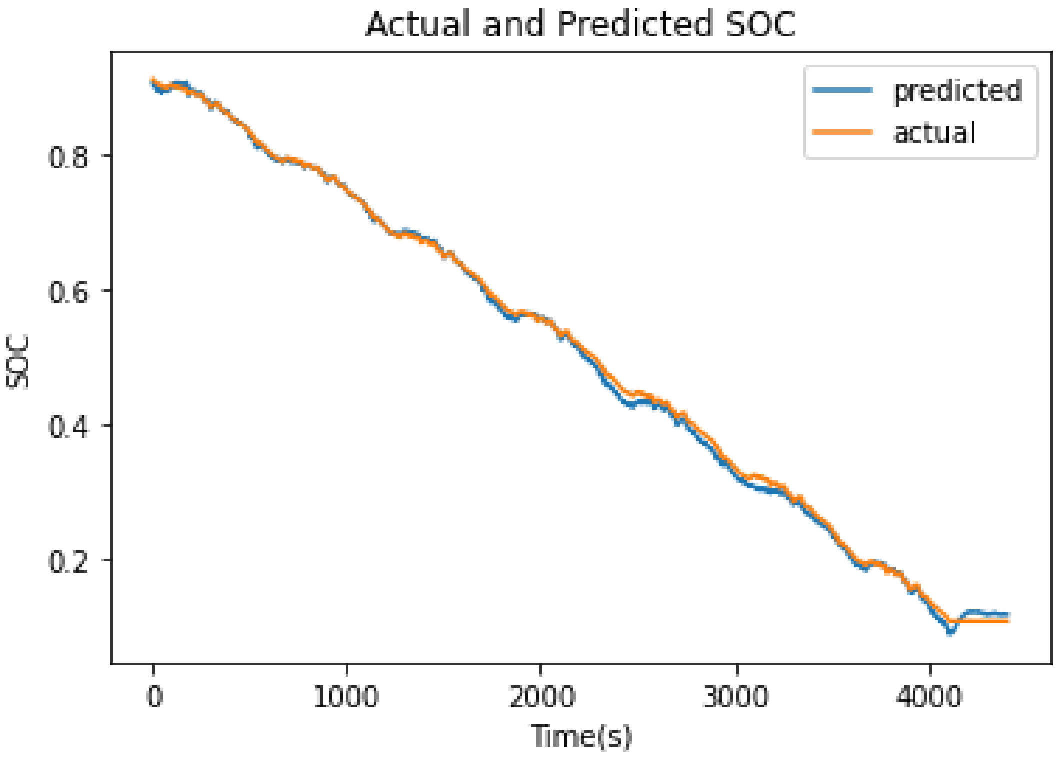


Figure 4.19: Base Models with dropout:Comparison of 3 feature and 4 feature model

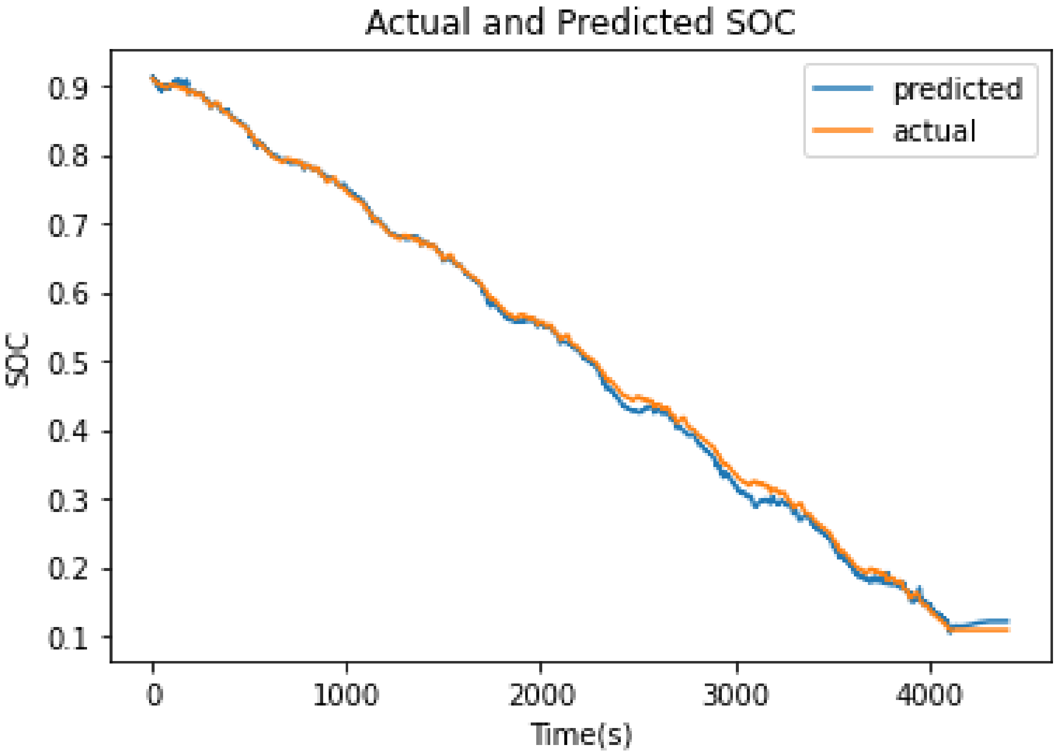
4.7.5 Full training: Base model with L1L2 and Dropout

In this section, base models were updated to include weight regularization through L1L2 and with the addition of the dropout layer. Best L1L2 and dropout for 3 feature models were: L1=0.0,L2=0.001, dropout rate=0.0 and for 4 feature model they were, L1=0.0,L2=0.01, dropout rate=0.0. Updated models were trained 10 times with Adam optimizer for 500 epochs.

Figure 4.20 and figure 4.21 represent the performance of trained models at 25°C and at 10°C for the US06 test drive cycle respectively by plotting actual and predicted SOC. Figure 4.22 and figure 4.23 represent the percentage error during same drive cycle at 25°C and 10°C respectively.

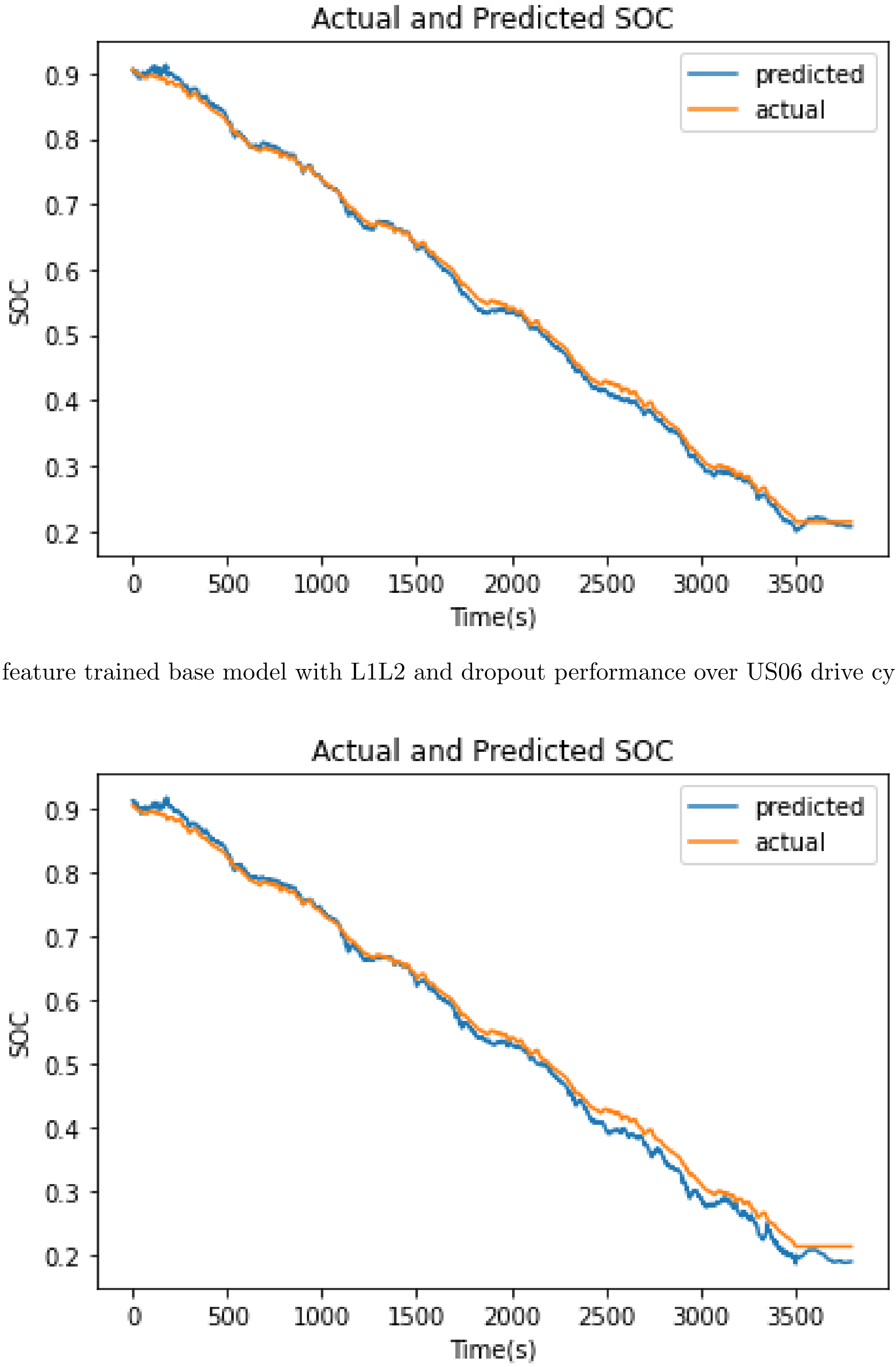


(a) 3 feature trained base model with L1L2 and dropout performance over US06 drive cycle at 25°C



(b) 4 feature trained base model with L1L2 and dropout performance over US06 drive cycle at 25°C

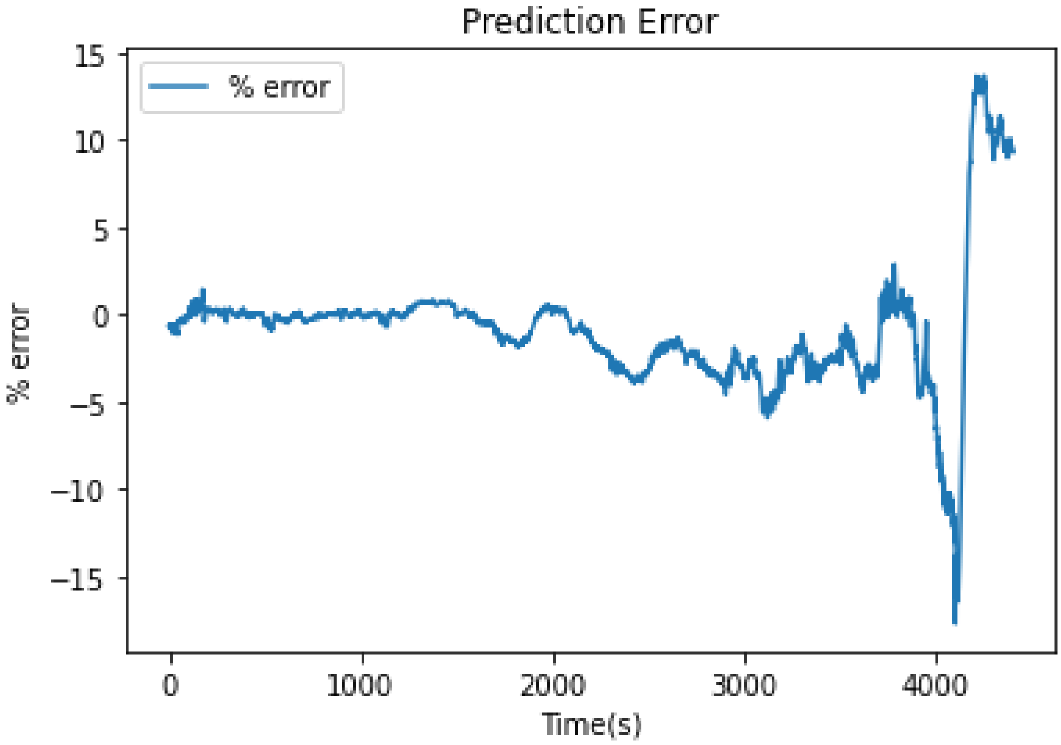
Figure 4.20: Trained base models with L1L2 and dropout at 25°C



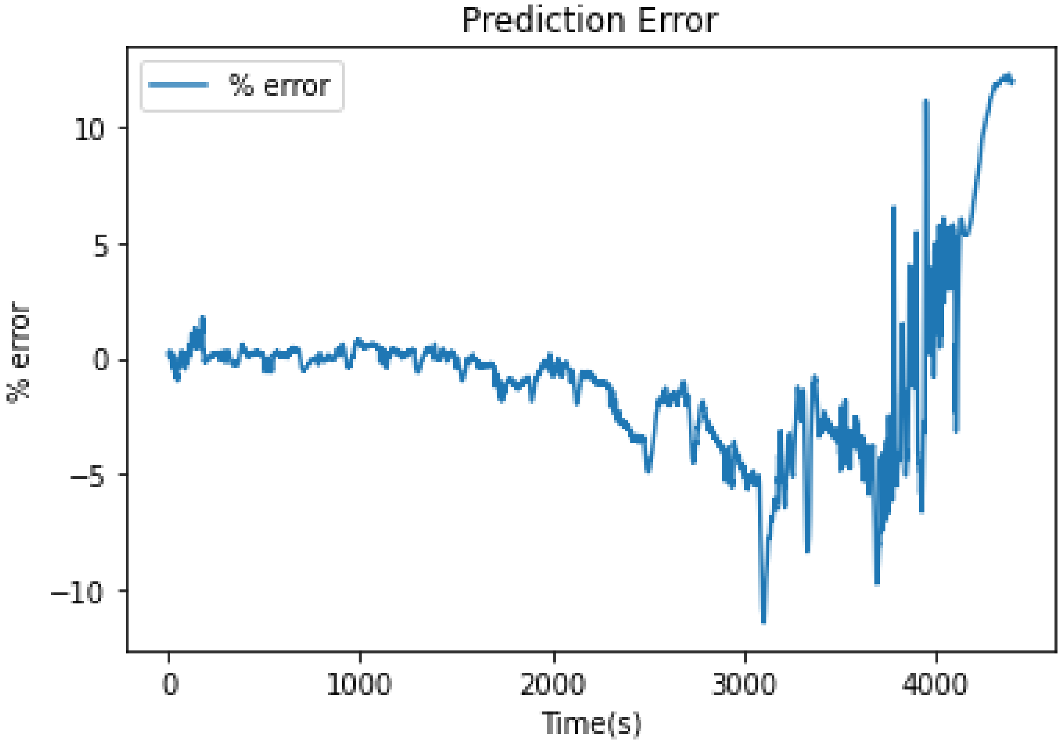
(a) 3 feature trained base model with L1L2 and dropout performance over US06 drive cycle at 10°C

(b) 4 feature trained base model with L1L2 and dropout performance over US06 drive cycle at 10°C

Figure 4.21: Trained base models with L1L2 and dropout at 10°C

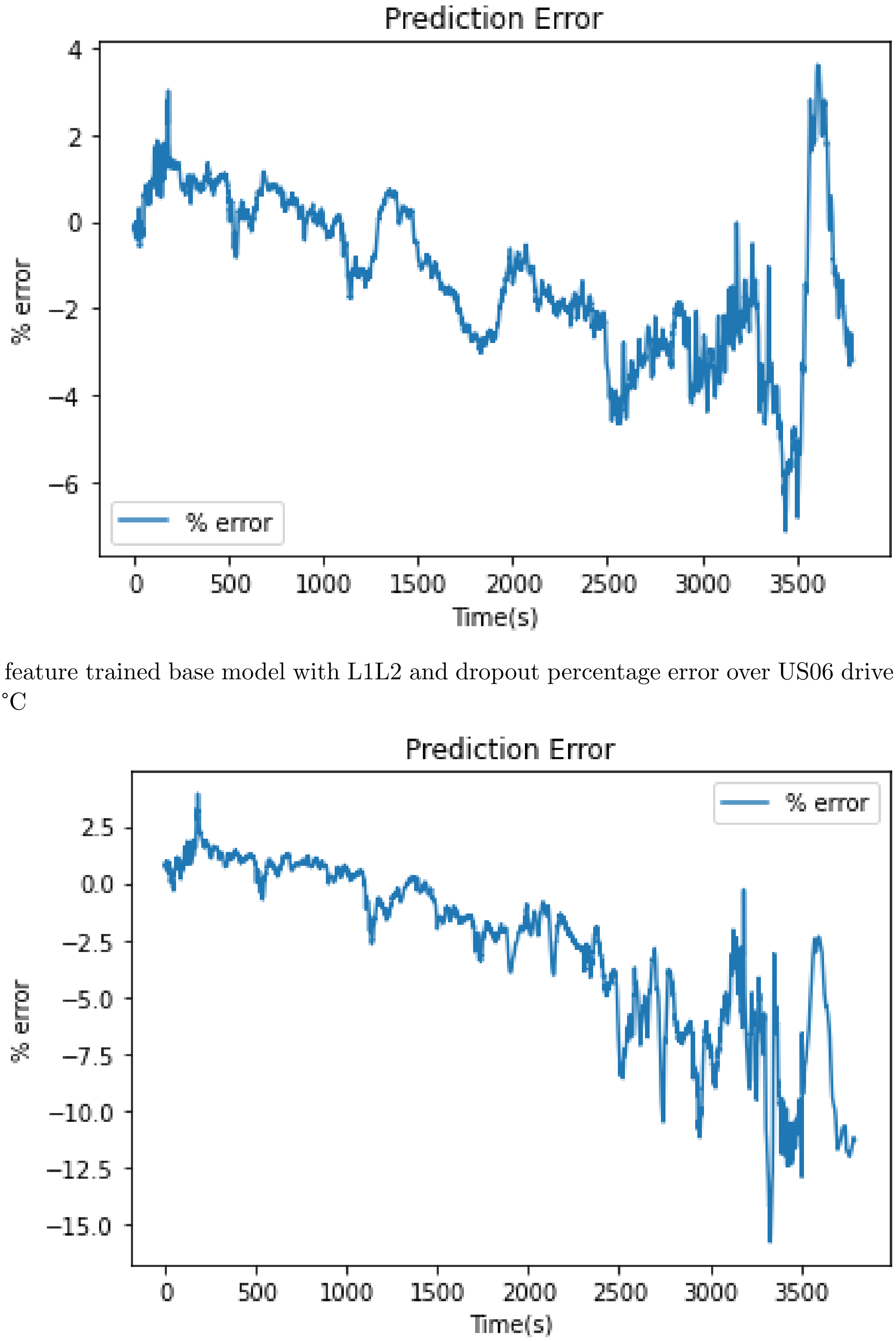


(a) 3 feature trained base model with L1L2 and dropout percentage error over US06 drive cycle at 25°C



(b) 4 feature trained base model with L1L2 and dropout percentage error over US06 drive cycle at 25°C

Figure 4.22: Trained base models with L1L2 and dropout percentage error at 25°C



(a) 3 feature trained base model with L1L2 and dropout percentage error over US06 drive cycle at 10°C

(b) 4 feature trained base model with L1L2 and dropout percentage error over US06 drive cycle at 10°C

Figure 4.23: Trained base models with L1L2 and dropout percentage error at 10°C

Figure 4.24 represents the comparison of 3 feature and 4 feature trained models over all test drive cycles. In this training, 3 feature models performed better for all test drive cycles. An extra feature in the 4 feature model did not add any value.

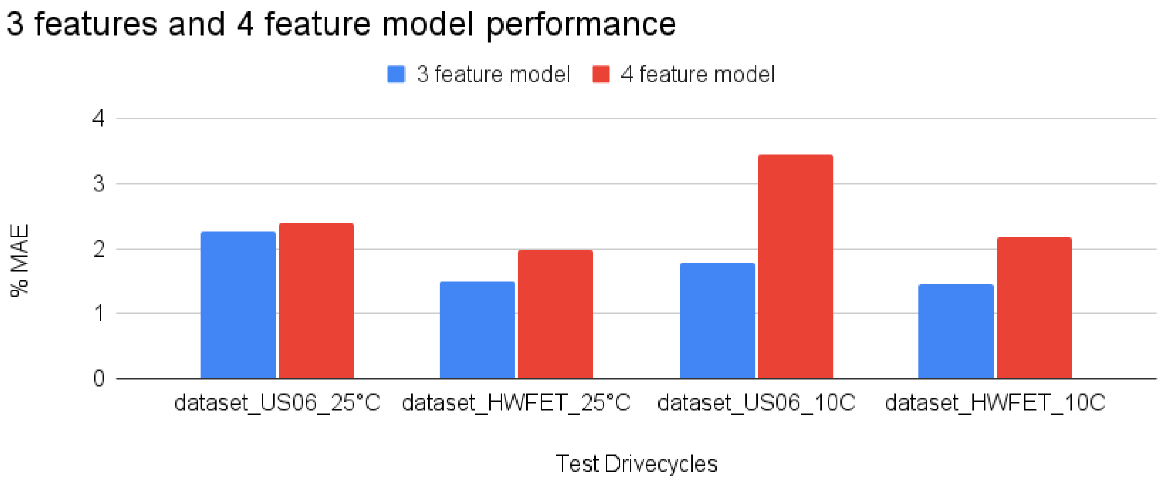


Figure 4.24: L1L2 and dropout:Comparison of 3 feature and 4 feature model

CHAPTER 5 CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

Machine learning based SOC estimation models are gaining a lot of interest due to their accuracy, flexibility, and agnostic nature to battery chemistry. Common inputs for these SOC estimation models are Terminal Voltage, Current being discharged/charged, and Temperature. In this work, LSTM-RNN based machine learning models were developed and trained. These models use Terminal Voltage, Current, and Surface temperature to model the battery behavior and produce State of Charge as output.They were able to predict SOC for real driving conditions in an excellent manner with limited training. In addition, a hypothesis is tested, if the SOC estimation model is supplied with information about the Voltage difference between current and past instances, it should improve the model performance. Since Voltage is strongly correlated with SOC, change in Voltage should affect change in SOC provided other parameters (Current and Temperature) are relatively similar. To supply additional information about Voltage change, an additional feature is added in the model: Voltage difference. So, now two categories of models exist 3 feature models with Voltage, Temperature, and Current as input features and 4 feature models with an additional feature of the Voltage difference. Both 3 feature and 4 feature models are trained and validated using identical data and their performance is evaluated. The following types of models were trained and their performance is summarized in Table 5.1.

Model variant	Impact of model performance 4th feature
Base	better for 25°C drive cycles
Base with L1L2	poor for all drive cycles
Base with Dropout layer	marginally better for 1 out of 4 test drive cycles
Base with L1L2 and Dropout layer	poor for all drive cycles

Table 5.1: Result summary

5.1.1 Impact of Voltage difference as a feature

From the results obtained in Table 5.1 with 4 model variants, 4 feature model performed better only in selected cases: (i) Base Model configuration and drive cycles at 25°C. (ii) Base Model with dropout layer configuration performed marginally better for 1 out of 5 drive cycles. From the correlation study in Section 4.4, voltage difference as a standalone feature is not very well correlated with SOC. From the studies done in this work, it appears that when voltage difference, when combined with other 3 features(Voltage, Temperature, and Current), may not be able to provide very useful insight to better model Li-ion battery behavior. This result might mean that Voltage difference as a feature did not provide any new useful information and likely introduced noise in the modeling which resulted in relatively poor performance when compared with 3 feature model.

5.2 Future work

In this work, Voltage difference was assessed for SOC estimation models. For future work, this feature analysis can be extended with the following variations:

1. Using more diverse training and test data. For initial prove out and ensure consistent state of health during testing, experimental data includes only 25°C and 10°C ambient conditions, but varying ambient temperature conditions might be an interesting study to compare with the results observed in this studies.
2. In this work, LSTM models were used due to their ability to remember past information. Another alternative is to use simple Deep Feed-Forward Networks but with average values of the current and average value of Temperature for a past fixed duration as input features. More details about the construction of such models can be found in [40].
3. Both of the above mentioned models can be trained with some noise added data to assess their robustness against the sensor bias and variations in practical use scenarios. Also, the time span for voltage difference can be tweaked to explore the trends.
4. Another line of research is to tweak the existing LSTM model construction to see if Voltage difference as an additional feature can improve model performance. Some of the construction changes in LSTM models can be: Change Number of LSTM units and layers, Change Optimizer for ADAM to other optimizers such as Gradient Prop, Back propagation etc., Change number of past instance from 400 to another

value. It has been found that 100 to 500 past instances bring a good result[40].

REFERENCES

1. Market Research Future (MRFR, “Lithium-Ion Battery Market 2020— Key Features, Growth Drivers, Upcoming Trends with Top Company Profiles (LG C,” openPR.com, Jul. 20, 2020. https://www.openpr.com/news/2093364/lithium-ion-battery-market-2020-key-features-growth-drivers (accessed Sep. 26, 2021).
2. “Vehicle Fuel Economy and Greenhouse Gas Standards: Frequently Asked Questions.” Accessed: Sep. 26, 2021. [Online]. Available: https://sgp.fas.org- /crs/misc/R4520-4.pdf.
3. “The 2020 EPA Automotive Trends Report,” 2021. Accessed: Sep. 26, 2021. [Online]. Available: https://nepis.epa.gov/Exe/ZyPDF.cgi?Dockey- =P1010U6-8.pdf.
4. A. Emadi, Advanced Electric Drive Vehicles. New York: CRC Press, 2015.
5. E. Chemali, M. Preindl, P. Malysz, and A. Emadi, “Electrochemical and Electrostatic Energy Storage and Management Systems for Electric Drive Vehicles: State-of-the-Art Review and Future Trends,” IEEE Journal of Emerging and Selected Topics in Power Electronics, vol. 4, no. 3, pp. 1117–1134, Sep. 2016, doi: 10.1109/jestpe.2016-.2566583.
6. R. Hemmati and H. Saboori, “Short-term bulk energy storage system scheduling for load leveling in unit commitment: modeling, optimization, and sensitivity analysis,” Journal of Advanced Research, vol. 7, no. 3, pp. 360–372, May 2016, doi: 10.1016/j.jare.2016.02.002.
7. E. Reihani, S. Sepasi, L. R. Roose, and M. Matsuura, “Energy management at the distribution grid using a Battery Energy Storage System (BESS),” International Journal of Electrical Power Energy Systems, vol. 77, pp. 337–344, May 2016, doi: 10.1016/j.ijepes.2015.11.035.
8. “Crude Oil Prices - 70 Year Historical Chart,” Macrotrends.net, 2020. https:/-/www.macrotrends.net/1369/crude-oil-price-history-chart (accessed Sep. 26, 2021).
9. A. Cherp and J. Jewell, “The three perspectives on energy security: intellectual history, disciplinary roots and the potential for integration,” Current Opinion in

Environmental Sustainability, vol. 3, no. 4, pp. 202–212, Sep. 2011, doi: 10.1016-/j.cosust.2011.07.001.

10. C. Winzer, “Conceptualizing energy security,” Energy Policy, vol. 46, pp. 36–48, Jul. 2012, doi: 10.1016/j.enpol.2012.02.067.
11. T. Couture and Y. Gagnon, “An analysis of feed-in tariff remuneration models: Implications for renewable energy investment,” Energy Policy, vol. 38, no. 2, pp. 955–965, Feb. 2010, doi: 10.1016/j.enpol.2009.10.047.
12. W. Sierzchula, S. Bakker, K. Maat, and B. van Wee, “The influence of financial incentives and other socio-economic factors on electric vehicle adoption,” Energy Policy, vol. 68, pp. 183–194, May 2014, doi: 10.1016/j.enpol.2014.01.043.
13. N. Bullard, “This Is the Dawning of the Age of the Battery,” Bloomberg.com, Dec. 17, 2020. https://www.bloomberg.com/news/articles/2020-12-17/this-is-the-dawning-of-the-age-of-the-battery (accessed Sep. 26, 2021)
14. Major Automaker’s announcements for electric lineup

    “GM’s Path to an All-Electric Future — General Motors,” Gm.com, 2020. https://www.gm.com/electric-vehicles.html (accessed Sep. 26, 2021).

    B. Klayman, “Ford boosts EV spending, outlines 2030 sales targets, shares near 5-year high,” Reuters, May 26, 2021. https://www.reuters.com/business/sustainable-business/ford-boosts-ev-spending-aims-have-40-volume-all-electric-by-2030-2021-05-26/ (accessed Sep. 26, 2021).

    A. J. Hawkins, “Toyota will release 15 new electric vehicles by 2025,” The Verge, Apr. 19, 2021. https://www.theverge.com/2021/4/19/22391738/toyota-electric-vehicle-strategy-bz4x-concept-subaru (accessed Sep. 26, 2021).
15. Li, Y,”Ensemble Bias-Correction Based State of Charge Estimation of LithiumIon Batteries”, Iowa State University(2017).
16. P. Malysz, R. Gu, J. Ye, H. Yang, and A. Emadi, “State-of-charge and state-of health estimation with state constraints and current sensor bias correction for electrified powertrain vehicle batteries,” IET Electrical Systems in Transportation, p. 9 pp., 2016.

17. L. McCurlie, M. Preindl, and A. Emadi, “Fast Model Predictive Control for Redistributive Lithium-Ion Battery Balancing,” IEEE Transactions on Industrial Electronics, vol. 64, no. 2, pp. 1350–1357, Feb. 2017, doi: 10.1109/tie.2016.2611488.

18. M. U. Ali, A. Zafar, S. H. Nengroo, S. Hussain, M. J. Alvi, and H.-J. Kim, “Towards a Smarter Battery Management System for Electric Vehicle Applications: A Critical Review of Lithium-Ion Battery State of Charge Estimation,” Energies, vol. 12, no. 3, p. 446, Jan. 2019, doi: 10.3390/en12030446.

19. W. Waag, C. Fleischer, and D. U. Sauer, “Critical review of the methods for monitoring of lithium-ion batteries in electric and hybrid vehicles,” Journal of Power Sources, vol. 258, pp. 321–339, Jul. 2014, doi: 10.1016/j.jpowsour.201- 4.02.064.

20. M.A. Roscher and D. U. Sauer, “Dynamic electric behavior and open-circuit voltage modeling of LiFePO4-based lithium ion secondary batteries,” Journal of Power Sources, vol. 196, no. 1, pp. 331–336, Jan. 2011, doi: 10.1016/j.jpowsour.2010.06.098.

21. J. Li, J. Klee Barillas, C. Guenther, and M. A. Danzer, “A comparative study of state of charge estimation algorithms for LiFePO4 batteries used in electric vehicles,” Journal of Power Sources, vol. 230, pp. 244–250, May 2013, doi: 10.1016/j.jpowsour.2012.12.057.

22. X. Chen, W. Shen, Z. Cao, and A. Kapoor, “A comparative study of observer design techniques for state of charge estimation in electric vehicles,” 2012 7th IEEE Conference on Industrial Electronics and Applications (ICIEA), Jul. 2012, doi: 10.1109/iciea.2012.6360705.

23. X. Hu, F. Sun, and Y. Zou, “Estimation of State of Charge of a Lithium-Ion Battery Pack for Electric Vehicles Using an Adaptive Luenberger Observer,” Energies, vol. 3, no. 9, pp. 1586–1603, Sep. 2010, doi: 10.3390/en3091586.

24. H. Chaoui and P. Sicard, “Accurate state of charge (SOC) estimation for batteries using a reduced-order observer,” 2011 IEEE International Conference on Industrial Technology, Mar. 2011, doi: 10.1109/icit.2011.5754342.

25. L. Liu, L. Y. Wang, Z. Chen, C. Wang, F. Lin, and H. Wang, “Integrated System Identification and State-of-Charge Estimation of Battery Systems,” IEEE Transacti-

ons on Energy Conversion, vol. 28, no. 1, pp. 12–23, Mar. 2013, doi: 10.1109/tec.2-012.2223700.

26. Y.-H. Chiang, W.-Y. Sean, and J.-C. Ke, "Online estimation of internal resistance and open-circuit voltage of lithium-ion batteries in electric vehicles," Journal of Power Sources, vol. 196, no. 8, pp. 3921–3932, Apr. 2011, doi: 10.1016/j.jpowsour.2011.01.005.
27. M. Gholizadeh and F. R. Salmasi, "Estimation of State of Charge, Unknown Nonlinearities, and State of Health of a Lithium-Ion Battery Based on a Comprehensive Unobservable Model," IEEE Transactions on Industrial Electronics, vol. 61, no. 3, pp. 1335–1344, Mar. 2014, doi: 10.1109/tie.2013.2259779.
28. X. Chen, W. Shen, Z. Cao, and A. Kapoor, "A novel approach for state of charge estimation based on adaptive switching gain sliding mode observer in electric vehicles," Journal of Power Sources, vol. 246, pp. 667–678, Jan. 2014, doi: 10.1016/j.jpowsour.2013.08.039.
29. T. Kim, W. Qiao, and L. Qu, "Online state of charge and electrical impedance estimation for multicell lithium-ion batteries," 2013 IEEE Transportation Electrification Conference and Expo (ITEC), Jun. 2013, doi: 10.1109/itec.2013.657-4523.
30. I.-S. Kim, "The novel state of charge estimation method for lithium battery using sliding mode observer," Journal of Power Sources, vol. 163, no. 1, pp. 584–590, Dec. 2006, doi: 10.1016/j.jpowsour.2006.09.006.
31. X. Chen, W. Shen, Z. Cao, and A. Kapoor, "A comparative study of observer design techniques for state of charge estimation in electric vehicles," 2012 7th IEEE Conference on Industrial Electronics and Applications (ICIEA), Jul. 2012, doi: 10.1109/iciea.2012.6360705.
32. Fei Zhang, Guangjun Liu, and Lijin Fang, "A battery state of charge estimation method using sliding mode observer," 2008 7th World Congress on Intelligent Control and Automation, 2008, doi: 10.1109/wcica.2008.4593055.
33. X. Chen, W. Shen, Z. Cao, and A. Kapoor, "Adaptive gain sliding mode observer

for state of charge estimation based on combined battery equivalent circuit model," Computers Chemical Engineering, vol. 64, pp. 114–123, May 2014, doi: 10.1016/j.compchemeng.2014.02.015.

34. C. Hu, B. D. Youn, and J. Chung, "A multiscale framework with extended Kalman filter for lithium-ion battery SOC and capacity estimation," Applied Energy, vol. 92, pp. 694–704, Apr. 2012, doi: 10.1016/j.apenergy.2011.08.002.

35. F. Sun, X. Hu, Y. Zou, and S. Li, "Adaptive unscented Kalman filtering for state of charge estimation of a lithium-ion battery for electric vehicles," Energy, vol. 36, no. 5, pp. 3531–3540, May 2011, doi: 10.1016/j.energy.2011.03.059.

36. S. Sepasi, R. Ghorbani, and B. Y. Liaw, "Improved extended Kalman filter for state of charge estimation of battery pack," Journal of Power Sources, vol. 255, pp. 368–376, Jun. 2014, doi: 10.1016/j.jpowsour.2013.12.093.

37. G. L. Plett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs," Journal of Power Sources, vol. 134, no. 2, pp. 252–261, Aug. 2004, doi: 10.1016/j.jpowsour.2004.02.031.

38. G. L. Plett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs Part 2. Modeling and identification," Journal of Power Sources, vol. 161, no. 2, pp. 1356–1368, 2004.

39. G. L. Plett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs," Journal of Power Sources, vol. 134, no. 2, pp. 277–292, Aug. 2004, doi: 10.1016/j.jpowsour.2004.02.033.

40. E. Chemali, "Intelligent State-of-Charge and State-of-Health Estimation Framework for Li-ion Batteries in Electrified Vehicles using Deep Learning Techniques," Mcmaster.ca, 2018, doi: http://hdl.handle.net/11375/23021.

41. Weights and Biases," AI Wiki, 2021. https://docs.paperspace.com/machinelearning/wiki/weights-and-biases (accessed Sep. 26, 2021).

42. J. Van Der Zwaag, "Using Domain-Specific Basic Functions for the Analysis of Supervised Artificial Neural Networks." [Online]. https://ris.utwente.nl/ws/files/6042013/thesis

43. “Hyperbolic Tangent,” Wolfram.com, 2021. https://mathworld.wolfram.com/HyperbolicTangent.html (accessed Sep. 26, 2021).

44. “A Gentle Introduction to the Rectified Linear Unit (ReLU),” Machine Learning Mastery, Jan. 08, 2019. https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/ (accessed Sep. 26, 2021).

45. Kingma, Diederik P and J. Ba, “Adam: A Method for Stochastic Optimization,” arXiv.org, 2014. https://arxiv.org/abs/1412.6980 (accessed Sep. 26, 2021).

46. Olah, C. (2015, August 27). Understanding LSTM networks. Understanding LSTM Networks – colah’s blog. Retrieved September 26, 2021, from https://colah.github.io/posts/2015-08-Understanding-LSTMs/.

47. P. J. Kollmeyer, ”Development and Implementation of a Battery-Electric Light-Duty Class 2a Truck Including Hybrid Energy Storage.” Order No. 3741528, The University of Wisconsin - Madison.

48. P. J. Kollmeyer, W. Lamb, L. W. Juang, J. D. McFarland, T. M. Jahns and B. Sarlioglu, ”Design of an electric powertrain for a Ford F150 crew cab truck utilizing a lithium battery pack and an interior PM synchronous machine drive,” 2012 IEEE Transportation Electrification Conference and Expo (ITEC), 2012, pp. 1-8, doi: 10.1109/ITEC.2012.6243511.

49. “How to Calculate Correlation Between Variables in Python,” Machine Learning Mastery, Apr. 26, 2018. https://machinelearningmastery.com/how-to-use-correlation-to-understand-the-relationship-between-variables/ (accessed Sep. 26, 2021).

50. E. Chemali, P. J. Kollmeyer, M. Preindl, and A. Emadi, “State-of-charge estimation of Li-ion batteries using deep neural networks: A machine learning approach,” Journal of Power Sources, vol. 400, pp. 242–255, Oct. 2018, doi: 10.1016/j.jpowsour.2018.06.104.

51. DeepAI, “Hyperparameter,” DeepAI, May 17, 2019. https://deepai.org/machine-learning-glossary-and-terms/hyperparameter (accessed Sep. 26, 2021).

52. “How to Reduce Overfitting With Dropout Regularization in Keras,” Machine Learning Mastery, Dec. 04, 2018. https://machinelearningmastery.com/how-to-

reduce-overfitting-with-dropout-regularization-in-keras/ (accessed Sep. 26, 2021).

ABSTRACT

ANALYZING A NEW FEATURE FOR STATE OF CHARGE ESTIMATION OF LI-ION BATTERIES

by

DEVANG MARVANIA

December 2021

Advisor: Dr. Leela Arava

Major: Mechanical Engineering

Degree: Master of Science

Accurate and reliable State estimation of Li-ion Batteries is paramount to electric vehicle safety and reliability. In this work, the focus is on developing Long Short Term Memory - Reinforced Neural Network(LSTM-RNN) models for State of Charge estimation and assessing whether an additional new feature can aid in State of Charge (SOC) estimation.

The most common features being used to predict SOC for Li-ion batteries are Voltage, Temperature, and Current. In this work, a hypothesis is tested: If a machine learning model is provided with additional information about how cell voltage is changing, it should improve model performance. This additional information is provided in the form of a feature: Voltage Difference. To test the hypothesis, 3 feature and 4 feature-based LSTM-RNN models are trained and tested. 3 feature models use Voltage, Current, and Temperature, while 4 feature models use an additional feature: Voltage Difference. First, a set of base models is created in 3 features and 4 features. Base models are trained using the same train data and performance is evaluated using the same test data. Next, hyper-parameters are introduced in base models. A grid search is performed among the possible values of hyperparameters to find the best value of hyperparameters. Models are updated with the best value of hyperparameter and trained and evaluated. Finally, a conclusion is made about whether the Voltage difference helped to improve the model performance or not.

AUTOBIOGRAPHICAL STATEMENT

Devang Marvania received his B.S. degree in Mechanical Engineering from S.V.National Institute of Technology,Surat(India) in 2016. He is pursuing the M.S. degree in Mechanical engineering from Wayne State University, Detroit. He is currently working at Gentherm's Advanced Engineering group to develop the next generation passenger climate comfort systems. Before joining Gentherm, he worked at Stellantis(previously Fiat Chrysler Automobiles) and Hutchinson Fluid Management Systems in electrified powertrain calibration and development respectively. Devang is passionate about using Robotics and Machine Learning to solve the pressing problems of modern times. He can be reached at devang.marvania@gmail.com to exchange ideas.