Tampere University

Bipin Lamichhane

ARTIFICIAL NEURAL NETWORK BASED STATE OF CHARGE ESTIMATION FOR LITHIUM-ION BATTERIES

Master's Thesis
Faculty of Information Technology and Communication Sciences (ITC)
Examiners: Professor Sami Repo, Postdoctoral Research Fellow Toni Simolin
June 2024

ABSTRACT

Bipin Lamichhane: Artificial Neural Network based State of Charge Estimation for Lithium-ion Batteries
Master's Thesis
Tampere University
Master's Degree Program in Electrical Engineering
June 2024

Lithium-ion batteries dominate the battery industry due to their high energy density, long life cycle, zero maintenance, minimal self-discharge rate, and versatility. These attributes have significantly advanced the green revolution, particularly through their use in Electric Vehicles (EVs) and Battery Energy Storage Systems (BESS). However, lithium-ion batteries are prone to voltage fluctuation and tend to overheat during charging. Lithium-ion batteries are also highly flammable and can undergo thermal runaway as they age. To oversee the overall operation of a lithium-ion battery during its operational life cycle, a Battery Management System (BMS) is used. The cost and complexity of BMS are dependent on its scope of applications, available features and don't have a unique set of criteria. The BMS monitors parameters like cell voltage, current, and module temperatures in-order to estimate the batteries' State of Charge (SOC) and State of Health (SOH). These parameters are of prime importance in the modern EVs to gauge the battery health and the range of EV. Since there is no direct measurement of SOC and SOH, they are estimated using mathematical techniques like coulomb counting, open circuit voltage, impedance measurement, etc. These techniques will require precise knowledge of battery composition. Modern EVs use lookup table and current counting method to estimate the SOC and ultimately the range of the vehicle. In certain contexts, these estimations demonstrate satisfactory performance, yet they are susceptible to errors and exhibit noise. However, the global demand for lithium-ion batteries is ever increasing and thousands of data points are measured all around the world by battery manufacturers, EV companies and most importantly, by users. These data points can be collected and made into a global battery database which can then be used to train Machine Learning (ML) and Artificial intelligence (AI) models. These models then can be used to estimate the SOC and SOH and other battery parameters that will help the BMS to make smart decisions during operation. This thesis explores the application of deep learning algorithms in the design of Battery Management Systems (BMS) integrated with artificial intelligence (AI). Two ANNs, Deep Neural Network (DNN) and Long Short Term Memory (LSTM) Network, estimated SOC of an LG 18650 HG2 cell at four temperatures: -10 °C, 0 °C, 10°C, and 25°C. Initially, DNN trained with voltage, current, and temperature as inputs yielded unsatisfactory results. Incorporating average voltage and average current along with those inputs, derived from 500 previous time-steps, significantly enhanced DNN performance, achieving a mean absolute error of (MAE) of approximately 0.99%. Further increasing neuron count and training epochs slightly improved performance to 0.97% MAE. LSTM, a type of recurrent neural network, was then utilized which is capable of bidirectional information flow and memory retention. Despite training for less number of epochs (200 epochs) compared to DNN (1000), LSTM exhibited competitive performance with an average MAE of 1.08%.

Keywords: Lithium ion batteries, Battery Management System, State of Charge (SOC), SOC estimation, Neural Networks, AI, Deep Neural Network, Long Short-Term Memory Network

ACKNOWLEDGMENT

This thesis would have been incomplete without the proper support and guidance of various individuals throughout the journey. I would like to take this time to remember them all and thank them for their constant support and love.

First and foremost, my heartfelt appreciation goes to my parents, whose unwavering belief in my abilities has been my constant anchor.

I extend my deepest gratitude to esteemed professor Dr. Sami Reppo for his supervision and critique on the thesis. I would also like to acknowledge the contributions of all other teachers and mentors who have been the architects of my intellectual growth.

I would also like to express my gratitude to Dr. Tuomas Messo who generously shared his insights, expertise, and valuable time. I am also profoundly thankful to the team at Cellife for their collaboration, support, and provision of resources during the course of my thesis work. This thesis is dedicated to all of them.

To all those who have been part of this academic journey, whether through encouragement, mentorship, or shared experiences, I extend my deepest regards.

This thesis stands as a testament to the collective effort of a community dedicated to the pursuit of knowledge and academic excellence. As this work is shared with the wider academic community, I hope it serves not only as an evidence to my personal growth but also as a contribution to the broader dialogue within the field. May it inspire future scholars and researchers to dive into the field of inquiry with the same determination that fueled this endeavor.

CONTENTS

LIST OF FIGURES

GLOSSARY

ANN	Artificial Neural Network
BESS	Battery Energy Storage System
BMS	Battery Management System
ECM	Equivalent Circuit Model
EIS	Electrochemical Impedance Spectroscopy
EOL	End of Life
EPA	United States Environmental Protection Agency
EVs	Electric Vehicles
GWh	Giga Watt hour
IC	Internal Combustion
LFP	Lithium Iron Phosphate
Li-ion	Lithium-ion
LIB	Lithium Ion Batteries
NMC	Nickel Manganese Cobalt
OCV	open Circuit Voltage
RNN	Recurrent Neural Network
RUL	Remaining Useful Life
SOC	State of Charge
SOH	State of Health
TAU	Tampere University
TUNI	Tampere Universities
Wh	Watt hour

1. INTRODUCTION

Lithium-ion (hereafter referred to as Li-ion) batteries have become ubiquitous in the modern world. Since their introduction in the early 1990s, they have been used in numerous applications due to their high energy density and long cycle life. Sony Corporation introduced the first commercial Li-ion batteries into the market and their popularity peaked with the increase in consumer electronics [59]. With the rising concern for global warming, the use of Li-ion batteries for the electrification of vehicles and as an energy storage system has gained a lot of research interest in the past decades. The urge for clean energy to reduce carbon footprints makes energy storage paramount in the future. This is because unlike non-renewable sources, energy from renewable sources like wind and solar are intermittent and require energy storage systems for continuous and efficient operation. At present times, different classes of energy storage medium are being researched for their performance and viability. Zakeri et al. [70] provides a list of energy storage systems and their life cycle cost assessments. Among them, batteries are one of the promising forms of energy storage medium because of their rapid response time, flexibility and short construction cycle. Currently, research is ongoing for numerous battery chemistries and cell types like lead acid, Li-ion, Sodium-Sulfur (NaS), metal air (KOH), and flow batteries. Lead acid is the oldest and the most mature technology which is currently used for power applications albeit being slowly replaced by Li-ion cells. However, out of all the battery chemistries Li-ion shows the greatest prospect for future development due to their high energy density and storage efficiency (almost 100%)[19]. Grid-connected Battery Energy Storage Systems (BESS) are one of the potential candidates for future smart grid systems, providing features like voltage stabilizing for grid-connected photovoltaic plants, load shifting, peak-demand supply, grid stabilization during supply-demand imbalance, etc [57]. For instance, AES Energy Storage Company at the Laurel Mountain Wind Farm, in Moraine, Ohio have built a Li-ion energy storage system with a peak power rating of 32 MW and energy storage capacity of 8MWh [70]. Presently, the largest Li-ion storage facility in the world is the Moss Landing Energy Storage System in Monterey County, California with a peak power rating of 750 MW and energy storage capacity of 3000 MWh [14]. These projects have proven the viability of the Li-ion chemistry in the energy storage systems for large scale applications like electricity grids.

The United States Environmental Protection Agency (EPA) stated that the transportation

sector generates the largest share of greenhouse gas emissions at around 28% (in 2021) [63]. Another article from International Energy Agency (IEA) stated that 24% emission in 2018 was by transportation sector out of which 45% is contributed by passenger vehicles [51]. Hence, an alternative mode of transportation or a new fuel source is imperative for reducing the carbon footprint generated by internal combustion engine vehicles. Currently, the two best alternatives to fossil fuel vehicles are the battery electric vehicles and the hydrogen fuel cell vehicles. CE Thomas [61] states that for any vehicles with range greater than 160 km (100 miles), hydrogen fuel cell vehicles are superior than battery EVs in terms of mass, volume, initial greenhouse gas reduction and life-cycle costs. However, hydrogen research is still in its nascent stage and would take some time to be a fully functional fuel sources. Besides, the high price of fuel cell vehicles, the lack of hydrogen infrastructure, high cost of building hydrogen refueling centers (compared to EV charging stations), high cost of hydrogen production, its storage and transportation have made it difficult for the world-wide adoption of fuel cell vehicles in the current times.

Alternatively, battery electric vehicles have improved drastically in the past decades providing high reliability and efficiency. Intelligent Battery Management Systems (BMS) are available to track the State of Charge (SOC) and State of Health (SOH) of the batteries. Fast charging stations are being built in large numbers around the world to support easy transition to battery electric vehicles. Because of such reasons, battery electric vehicles are dominating the alternative transportation landscape, offering unparalleled efficiency, reduced emissions, and a sustainable solution to address the pressing challenges of climate change and environmental degradation. This has triggered remarkable efforts in the field of EV manufacturing and design. Similar to passenger cars, other mode of transportation that can benefit from battery electrification are short distance ferries, delivery trucks and mobile working machines like small excavators, mining equipment like drills etc. Batteries pose a problem in situations where vehicle weight must be controlled, such as in airplanes and cargo ships. This is probably one of the reasons why electrification of aeroplanes are not considered practical.

The global rise of Electric Vehicles (EVs) is one of the driving factors for bulk research in battery industry as batteries are paramount in EV manufacturing.The electric car market showed exponential growth as sales exceeded 10 million in 2022. The global share of EVs in total sales has tripled in three years, from around 4 % in 2020 to 14% in 2022 [1]. This has also prompted numerous research and advancements in battery technologies. By 2040, the European Commission projects about 150 to 900 million electric vehicles and 1300 GWh stationary storage. According to the key projections, the global sales of Li-ion batteries will increase to around 4 TWh [15] in the high, 2 TWh [15] in the moderate, and 0.6 TWh [15] in the low scenario by 2040 [15].With this increase in electric vehicles and grid-level storage, Li-ion batteries are expected to be more cost-effective and reach the market in great numbers. The increase in demand for electric vehicles is also driving

the demand for Li-ion batteries. Automotive Li-ion battery demand increased by about 65% to 550 GWh in 2022 from about 331 GWh in 2021 [1]. These figures show that Li-ion technology is the future of battery energy storage. Due to their exceptionally high energy density ($\sim$ 300 Wh/kg) compared to other battery chemistries ($\sim$ 75Wh/kg), high power-to-weight ratio, and highly modular nature, they are the first choice as a power source in Electric Vehicles (EVs)[31].

EVs are efficient than traditional IC engine vehicles as 59-62% of energy they consume can be converted to power at the wheels compared to 17-20% in conventional IC engines.[39]. However, Li-ion battery packs in EVs have their own set of problems. The biggest of which is the estimation of remaining amount of charge which directly relates to the available range of the EV. The amount of charge remaining as a function of the nominal capacity of the battery is termed as State of Charge (SOC) of the battery. It is desirable to operate batteries at a certain SOC range for long life and optimum performance [37]. Hence, the knowledge of battery's SOC is important for estimating other parameters like State of Health (SOH), ageing, etc. However, estimating SOC is a difficult task as it cannot be measured directly. Traditionally, two approaches have been used to estimate the SOC of batteries: Coulomb Counting and Open Circuit Voltage (OCV). These simple approaches have been replaced by advanced estimation techniques like Luenberger observer, Adaptive observer, sliding mode observer and Kalman filters for increased accuracy. These methods usually apply cell model with adaptive parameters, to estimate the SOC. These methods require accurate and precise models to perform SOC estimation at varying ambient temperature [8] [16]. So, limited prior knowledge regarding the battery chemistry leads to poor model design. Thus, these approaches require extensive study to formulate accurate models of the battery for SOC prediction.

Compared to above mentioned estimation techniques based on Equivalent Circuit Model(ECM), data-driven approaches are less complex and simple in structure. The estimation of SOC based on data-driven approach is assumed to be counter intuitive as the operating conditions of batteries are complex and non linear. However, with the availability of powerful computing machines, large database and using multiple hidden layers, these non linearity between the input and output can be tackled with remarkable accuracy. Different deep learning algorithms like Artificial Neural Network (ANN), Long-short Term Memory (LSTM), Recurrent Neural Network (RNN), Fully Connected Neural Network (FCNN), Temporal Convolutional Network (TCN), Non Linear Auto-regressive Neural Network (NARX) etc. use hidden layers with non-linear activation functions like sigmoid and hyperbolic tangent to describe and model the non-linearity between inputs and outputs[25].

There are many neural network models that are currently under research for battery SOC estimation. However, this is difficult as there are myriads of battery parameters that come

at play when determining the SOC of a cell. Besides, neural network model need large set of data to train and they have risk of over-fitting if the parameters are not set properly. Which network parameters play the important role? Are the model able to handle non-linear relationship between battery physical parameters like voltage, current, temperature, age, chemistry etc. and SOC? How can the model be trained and what shall be done to increase the training efficiency? This thesis attempts to answer these questions. The objective of this thesis is to design a Artificial Neural Network(ANN) that estimates the State of Charge (SOC) of a Li-ion battery and analyze its performance and shortcomings. More specifically, this thesis deals with use of Feed Forward Deep Neural Network (FFDNN) and Long-Short Term Memory (LSTM) network to estimate the SOC of the Li-ion batteries. FFDNN is one of the simplest types of neural network architecture. The information flow is unidirectional (hence the name feed forward). It is simple to implement and has fewer complexities while modelling in programming environments. Hence, it has been chosen for the SOC estimation purpose. However, FFDNN are rudimentary for time series data analysis like SOC where the input from previous time step is helpful for estimating current time step value. This is where Long Short Term Memory (LSTM) network prove their superiority. LSTM networks have a feedback loop in them which uses information from previous time step to make estimations in the current time step. This network architecture is famous in time series data estimations like load forecasting, stock market predictions etc. Hence, it has been chosen as a suitable architecture to contrast performance against FFDNN. More information on these have been provided in section 3.

This thesis is organized as follows: Section 2 deals with the background of Li-ion batteries and the problem of SOC estimation. Section 3 deals with the implementation of a neural network model, estimation of SOC, and calculation of error metrics. Section 4 presents the analysis of the results obtained from the model. Section 5 dives deep into the discussion of the results and observations made from the entire process. The thesis is concluded in section 6 with references for future work.

2. THEORY OF LI-ION BATTERIES, BATTERY MANAGEMENT SYSTEM AND SOC ESTIMATION

Lithium batteries have been in the market since the early 1990's. The elemental Lithium is the lightest metal with the highest electrochemical potential and provides the highest energy density for the given weight making it suitable for portable electronics and EVs. Before delving deep into the estimation of SOC, a short introduction of Li-ion batteries and BMS is give here.

2.1 Lithium-ion Battery

Li-ion batteries are a category of rechargeable cells that use lithium ions as their energy storing element. A typical Li-ion cell consists of four main components as shown in figure 2.1:

- a positive electrode (cathode) made of lithium liberating compounds like Lithium Cobalt-Oxide ($LiCoO_2$), Lithium Manganese-Oxide ($LiMn_2O_4$), Lithium Nickel-oxide ($LiNiO_2$), Lithium Iron-Phosphate ($LiFePo_4$) etc. It is the main source of lithium ions and determines the capacity and the average voltage of the battery.
- a negative electrode (anode) made of carbon, usually graphite, that stores lithium ions in its crystal structure (termed as intercalation) and allows the releases of lithium ions from the cathode, allowing the pass of currents through the external circuits.
- a separator layer which is usually a porous membrane that prevents the short circuit between the electrodes and helps in transfer of ions. It is usually made of organic polymers like polyethylene (PE), polypropylene (PP) or PE/PP with pore size in the ranges of the micrometers. [38]
- and an electrolyte made of lithium salt in organic solvents like lithium hexafluorophosphate ($LiPF_6$) that works as a medium for lithium ions movements.

During Li-ion battery charging, energy from the external circuit drives lithium ions from the positive electrode to the negative electrode, where they intercalate with electrons into the graphite crystal structure. Full charge occurs when all lithium ions are intercalated into the graphite. During discharge, lithium ions move back to the positive electrode via

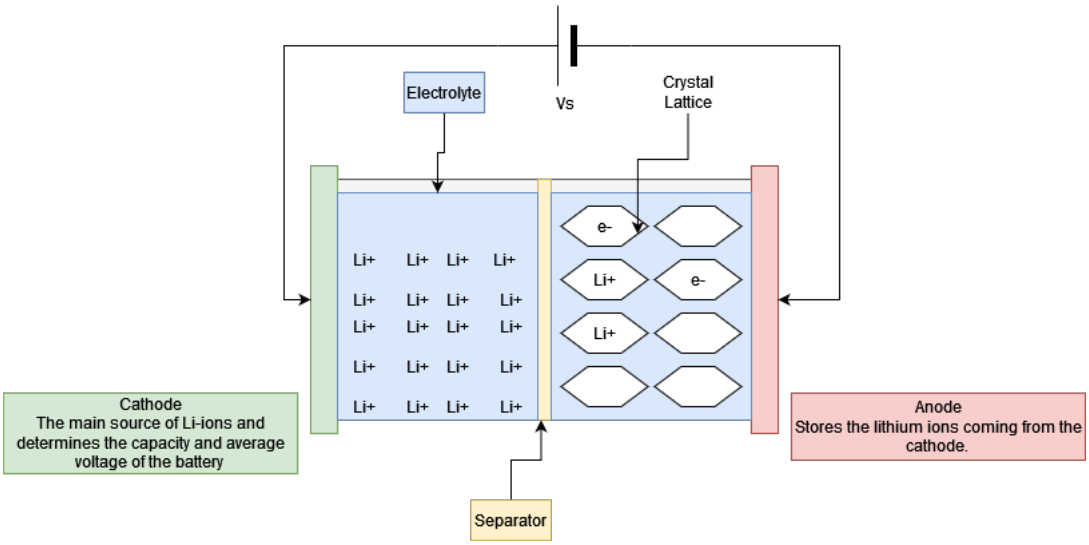

Figure 2.1. *Schematics of Li-ion Battery*

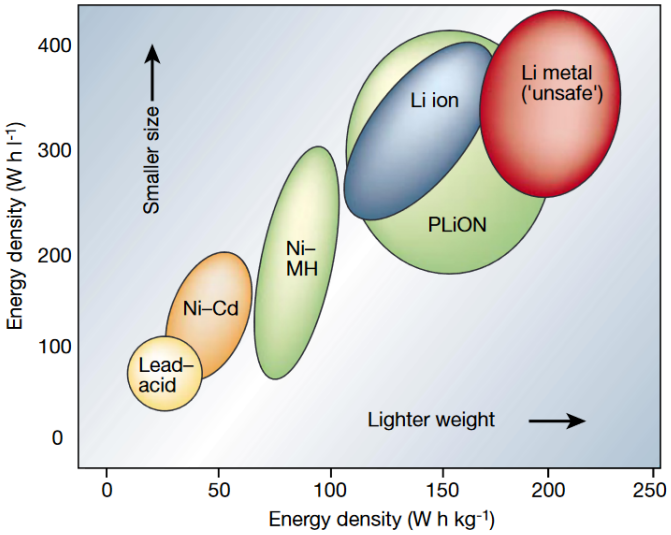

Figure 2.2. *Comparison of different cell chemistries. (Taken reference from [60])*

the separator and electrolyte, while electrons flow to the positive electrode through the external circuit. The separator permits Li-ion passage but blocks electron flow. Intercalation and deintercalation refer to the insertion and removal of Li-ions and electrons into the graphite structure, respectively, a process repeated during charge and discharge cycles. This cycling resemblance has led to the term "rocking chair batteries" for lithium batteries. The major focus on Li-ion batteries compared to other battery chemistries is due to the fact that Li-ion batteries are superior in most of the benchmark tests. Li-ion batteries provide high energy density, flexible and lightweight design, and a longer lifespan compared to other battery technologies.

Figure 2.2 compares cell chemistries and energy density. Li-ion boasts an energy density nearly 5 times higher than lead acid batteries [59], ideal for portable electronics and

electric vehicles due to its compactness. Its stable chemistry ensures longevity and faster recharge times, especially important with the rise of EVs [43]. However, safety concerns persist, including overheating risks and environmental impact due to scarce and toxic materials like lithium and cobalt. Recycling remains a challenge, with only 5% of Li-ion batteries recycled [56]. Moreover, Li-ion batteries are sensitive to overcharging, undercharging, and overheating, necessitating careful monitoring and management through a Battery Management System (BMS).

2.2 Battery Management System

Li-ion batteries packs are always equipped with a battery monitoring system also termed as BMS. A BMS is a combination of hardware and software units that continuously determine the battery state using the predefined algorithms and protocol. This process is called as battery monitoring [66]. BMS unit can be aptly termed as the brain of the battery energy system. BMS has features like charge control, battery efficiency computations, remaining run-time information, cycle counting, battery life prognosis, thermal management, prediction of battery failure, and safety alarm for over-the-limit usage [57]. The charging current and the temperature must be monitored for Li-ion batteries to operate within their safe limits, which is the main function of BMS. The BMS also helps in cell balancing and prevents cells from fully discharging. This helps the cells operate at their optimum level, increasing their operating life. A further aspect of BMS is the temperature monitoring of cells to prevent temperature extremes. This thermal control also helps batteries operate at their optimum performance and increases the effective operating life of the battery. A block diagram showing the major components of a typical BMS used in EVs is shown in figure 2.3.

As shown in figure 2.3, BMS has central control over different battery parameters. It provides charge control and monitoring. Besides, a cell module is made up of individual cells, that require cell balancing which is another function of BMS. However, the main feature of BMS is its safety mechanism. BMS safeguards battery from overcharging, over discharging and overheating. Modern BMS are also equipped with data control and data communication features. This makes them able to transmit information to the user for further analysis and easy decision making, for example in the case of EVs.

BMS acts on external as well as internal triggers. The electrochemical nature of the battery is complex and it performs differently in various operational and environmental conditions. BMS is tasked with measuring, logging, and sensing the battery parameters like cell voltage, discharge current, module temperature, ambient temperature, charge/discharge cycle counts, etc. BMS comes in different configurations based on its applications. Consumer electronics like hand-held and portable electronics have relatively simple BMS

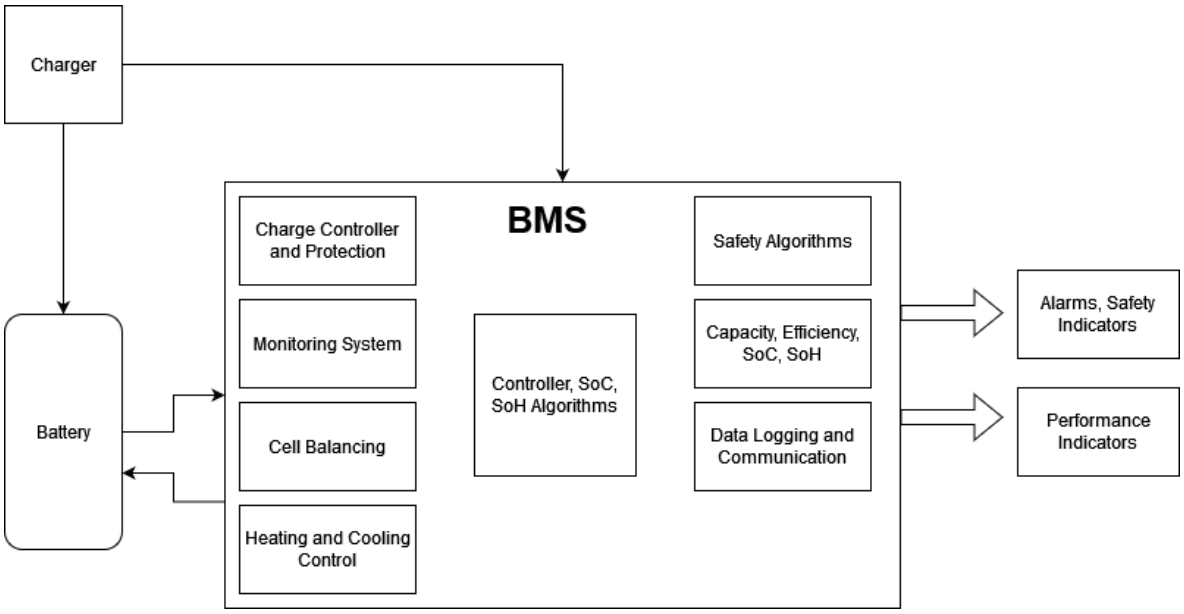

Figure 2.3. *Block diagram of a typical BMS*

compared to an EV. This is because cells used in EVs are hundreds of times bigger than those used in consumer electronics. They are designed for high-power applications like driving the inductor motor and hence should be able to handle high voltage, high current, and frequent charge/discharge cycles. This makes the design of BMS for EVs more complicated than those for portable electronics[68].

A BMS is a combination of hardware and software units. BMS has safety mechanisms in place for over-temperature and charge control. Charge control is necessary as Li-ion batteries use the Constant Current/Constant Voltage (CC/CV) charging principle. Hence, a galvanostat and a potentiostat are important hardware equipment in BMS [26]. A thermal module is present to sense temperature imbalance. Apart from these, data communication is necessary with the vehicle's onboard system. So, a transmitter and a receiver units are also present. Modern BMS are micro-processor driven and hence have data acquisition systems for data collection. A smart charge controller is also present to limit the charging and discharging current.

While the hardware interacts directly with the physical battery, most of the decision-making and analysis takes place using software. So, software can be aptly considered the "Brain of the BMS". Estimation of states, fault identification, making decisions, control of the hardware equipment, sensor control, switching control, etc. are all performed with software. Besides these, data analysis, online data manipulation, and data storage are other important tasks carried out via software. Cell balancing, temperature monitoring, and smoke detection are very important tasks that software handles. State estimation of battery also includes the estimation of SOC and SoH which are calculated based on various models and algorithms like state space model, neural network, fuzzy logic, etc. SOC is estimated using the voltage, current, and temperature measurement while SoH is esti-

mated by measuring performance degradation in the battery which is related to capacity fade and power fade [26].

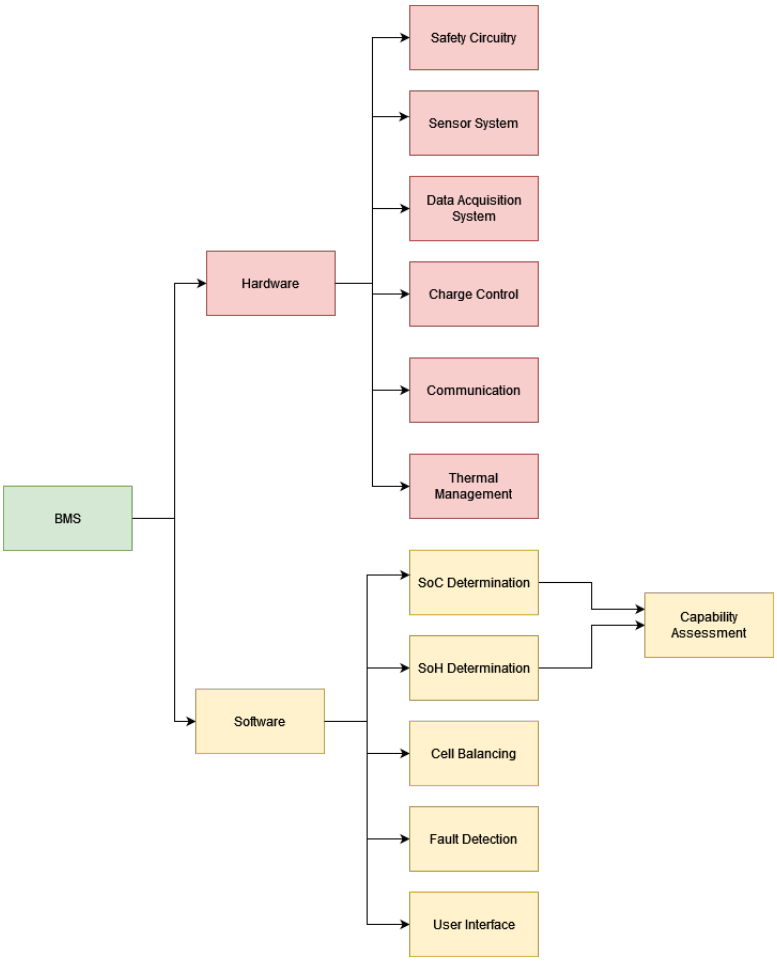

Figure 2.4. *Software and Hardware specifications for a BMS*

Figure 2.4 shows a basic breakdown of hardware and software units in a typical BMS [68]. The hardware consists of safety and data acquisition parts along with communication channels and thermal management system during overloads and overheating. The safety circuitry consists of the over-voltage and over-current protection. Usually a circuit breaker is used to trip the load or the charging circuit if the voltage/current is more than the recommended value. The sensor system is a complex network of voltage sensor, current sensor, temperature sensor among others. These feed the data to the processing unit for further processing and analysis like calculation of battery voltage, remaining time for charging/discharging etc vai the DAQ (Data Acquisition System). DAQ are ADC/DAC (Analogue to Digital Conversion and vice versa) capable device that converts the continuous analogue signals (like voltage and current) to digital signals for processing. Thermal management has to be done because of the thermal runaway problem in Li-ion batteries. As batteries age, there is significant heating during the charge/discharge cycle. Thus, the temperature of the batteries needs to be observed.

Software, on the other hand, is responsible for decision and calculations. Based on the current and voltage measurements, software calculates the capacity of the battery. It also tracks cell voltage level of individual cells in a module and charges/discharges them as necessary. If the measured values are indicators of any fault (for example: a battery showing low voltage even after charging), the software alerts the owner for checks and maintenance. Most importantly, BMS also provides, the capacity assessment of the battery which tells the owner how much charge is left (i.e SOC) and what's the age or health of the cell (i.e. SOH).

2.3 Estimation of State of Charge (SOC)

State of Charge is defined as the ratio of the available amount of charge in the battery to the maximum amount of charge the battery can hold when fully charged at the same specific standard condition [42]. It is usually expressed in percentage such that 0 % represents battery with no usable charge while 100 % represent battery that is fully charged. Mathematically it can be expressed as:

$$SOC = \frac{Q_{available}}{Q_{usable}} * 100\% \tag{2.1}$$

One thing to notice is that the usable charge of the battery is a little less than the rated charge of the battery due to internal heating loss, side reactions and aging. Estimation of SOC is a difficult task because it is not measurable directly and relies on mathematical formulation or other estimation methods. To add to the complexity, SOC is affected by battery aging as the electrochemical side reactions occurring in the battery reduces the maximum capacity of the battery over time. This leads to non-stationary, non-linear battery degradation characteristics. Besides, the temperature of the battery also affects the estimation of SOC. Despite these challenges, the measurement of SOC is critical as it directly relates to the available range in an EV. It acts as a foundation for estimating the State of Health (SoH), State of Power (SoP), Remaining Useful Life (RUL), Battery Age, Calendar Life, and so on. Accurate prediction of SOC gives an idea of charging/discharging strategies too. Especially in EVs, SOC is analogous to the fuel gauge in a traditional IC engine vehicles [26] [68]. Modern EVs use a combination of techniques mentioned below to draw out conclusion regarding the SOC of the battery. However, exact techniques are company secrets and each company deals with this problem independently. Accurately predicting SOC is an important milestone in developing reliable battery energy storage systems. Before moving onto the neural network models, conventional approaches which are used for estimation of SOC have been discussed below.

2.3.1 Coulomb Counting

Several mathematical methods have been introduced since 1980's to estimate the SOC of batteries. Among them, the most common method is coulomb counting. It uses the current integration to estimate the available capacity of the battery and predict the SOC. Mathematically, the coulomb counting method is expressed as:

$$SOC = 1 - \frac{\eta \int_{t_0}^{t} i \, dt}{C_n} \tag{2.2}$$

where $\int_0^t i \, dt$ represents, the integral of current over time period of t - t_0 sec and η represents a constant called Coulomb efficiency. Coulomb efficiency refers to the ratio of discharge capacity after full charge to the charging capacity of the same cycle. It represents how much energy during charging gets transferred while discharging and is a numerical value less than 1. Coleman et al. [13] define the SOC of the battery as shown:

$$\theta = \theta_0 - \frac{1}{C_N} \int_{t_0}^{t} \delta_{(I)} I_{\mathrm{Batt}} dt \tag{2.3}$$

where θ_0 is the rated SOC, C_N is the rated nominal capacity, I_{Batt} is the battery current, and $\delta_{(I)}$ is the current loss efficiency (usually 0.98-1). Theoretically, this is the most exact method of estimating the SOC of the battery. This method is simple to implement requiring low computational power. However, in reality there are issues with the accuracy of the value obtained using this approach. Estimating the initial rated capacity is a challenge in itself. The error in initial estimation of rated capacity of battery may have cumulative effect on the final value of predicted SOC. Hence, periodic re-calibration of measurement is necessary. Also, this is an open loop approach and can result in significant error due to disturbances like noise, temperature, etc. [26].

2.3.2 Open Circuit Voltage

Snihir et al.[55] used statistical method to predict the open-circuit voltage on the basis of voltage curves obtained by charging batteries with different currents. They found that if a battery is allotted enough resting time for charge balance, the Open Circuit Voltage (OCV) can be used to predict the SOC of batteries. However, the relationship between OCV and SOC is different for different types of batteries. The relationship depends on the material and capacity of the battery. For example, lead -acid cells have linear relationship between OCV and SOC while Li-ion cells don't [58] [26]. This method is simple and can be highly precise but it suffers from the drawback of having to wait long time for the charge balance to take place. For example in LFP cell, reaching this equilibrium can take as much as two hours [26]. This time lag is a prob-

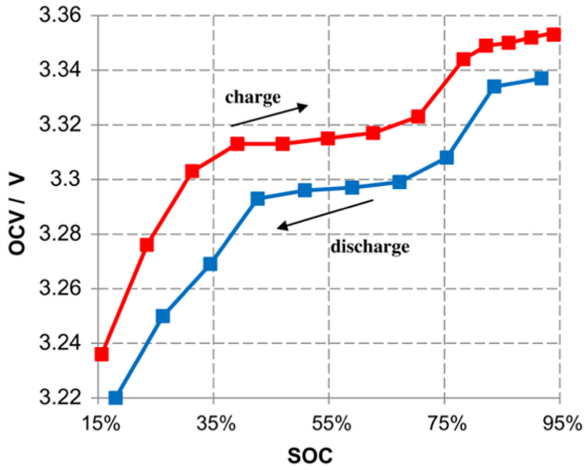

Figure 2.5. *OCV/V vs SOC charge and discharge profile of C/LiFePO4 battery tested under 25 °C, for 3 h [42]*

lem for EVs as waiting two hours for correct prediction of SOC is unrealistic in daily usage.

Also, batteries have hysteresis characteristics which results in low open circuit voltage during discharge and high open circuit voltage when charged as shown in figure 2.5. It can be observed from the figure that the open circuit voltage (OCV) is different for charging and discharging cases. The OCV is higher while charging. Besides, OCV is dependent on the temperature as well making it even harder to make predictions based on OCV alone. So careful consideration of voltage values is just as important.

2.3.3 Internal Resistance Measurement

This method takes into consideration the battery voltage and current to measure the internal resistance of the battery. Voltage is measured with the variation in current usually in the small time range of less than 10 ms [26]. The resistance so calculated is termed as DC resistance which represents the capacity of battery in DC. The DC resistance ranges in m$\Omega$and it is very hard to accurately obtain the value. Also, the internal resistance of battery changes slightly with the wide range of SOC values as shown in figure 2.6 which makes it difficult to predict SOC. Because of these reasons, this approach is almost never used for prediction of SOC.

2.3.4 Electrochemical Impedance Spectroscopy (EIS)

EIS is a powerful tool to understand the dynamics of the Li-ion cell. It works by applying small excitation current/voltage wave to the electrode and records the resulting response in the form of voltage and current. These perturbation signals are usually sinusoidal potential wave of amplitude E_{ac0} superimposed on a constant dc potential E_{dc0}. This

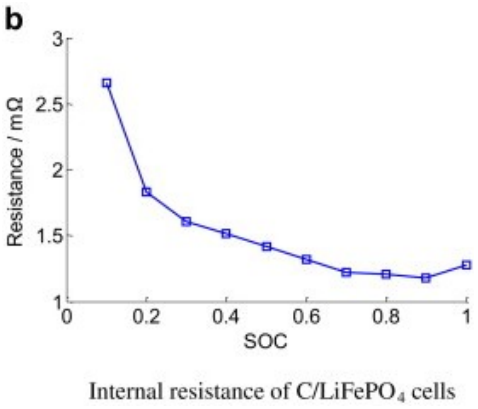

Figure 2.6. *Variation of internal resistance with respect to SOC in Li-ion batteries [42]*

gives a current response which can be written in Fourier series as:

$$i = i_0 + i_1 + \sin(\omega t + \Psi_1) + i_2 + \sin(2\omega t + \Psi_2) + ...i_n + \sin(n\omega t + \Psi_n) \qquad (2.4)$$

The magnitude (i_1) and phase (Ψ_1) of this current density are recorded. The vector ratio of the this potential to current is denoted as impedance [20]. By using excitation at different frequencies and at various SOCs, it is possible to develop accurate equivalent circuit models of the battery [40]. EIS uses these models to estimate the impedance at wide range of frequencies and develop an accurate model of the battery. A typical first order equivalent circuit model of impedance of a Li-ion battery is shown in figure 2.7.

After this, using the correlation between the obtained EIS model and SOC, this technique can be used to estimate the SOC of the battery. Kong et al. [36] have proposed a fast SOC estimation using this technique. They have stated that

"based on the correlation analysis between the solution-phase lithium-ion concentration, the solid-phase surface lithium-ion stoichiometry and the obtained electrochemical impedance spectroscopy (EIS) data, key frequency features for SOC estimation are extracted ".

The drawbacks of using EIS is that it only works for steady-state operation. This method is also susceptible to the influence of battery ageing and temperature variation giving inaccurate results [26]. Coleman et al. [13] have conducted experiments to measure the battery EMF voltage using impedance, terminal voltage and discharge current under load. It is also stated that using impedance alone as a function of SOC would give inaccurate results for Li-ion batteries [13].

2.3.5 Kalman Filter

Kalman filter is a recent development in battery prognosis that has been researched substantially in past decades. Unlike the conventional methods discussed before, it is an

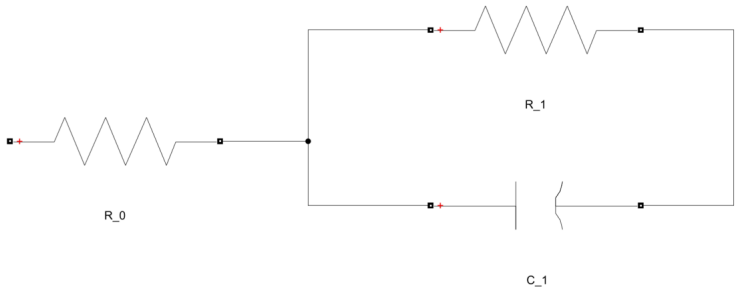

Figure 2.7. *First order equivalent circuit model of the impedance of Li-ion Battery*

adaptive algorithm that predicts new state in a dynamic system while correcting the new state repeatedly as the system operates. This is an intelligent algorithm that can be used to estimate the SOC of Li-ion battery. This method uses a set of mathematical equations that represent the cell dynamics and are written in state-space form as shown in the following equation that represents the dynamics of the lithium-ion cell in dynamic environment. This model is also called hysteresis model of the cell contrary to the simple model as explained in [44]

$$\begin{bmatrix} h_{k+1} \\ SOC_{K+1} \end{bmatrix} = \begin{bmatrix} F(i_k) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} h_k \\ SOC_k \end{bmatrix} + \begin{bmatrix} 0 & (1 - F(i_k)) \\ -\frac{\eta i_k \Delta t}{C_n} & 0 \end{bmatrix} \begin{bmatrix} i_k \\ M_k \end{bmatrix} \tag{2.5}$$

$$F(i_k) = exp(-(\frac{\eta i_k \Delta t}{C_n})) \tag{2.6}$$

$$V_k = OCV(SOC_k) - Ri_k + h_k \tag{2.7}$$

where η_i represent the columbic efficiency of the cell during charge and discharge R is the internal resistance of the cell and h_k represents the hysteresis characteristic of the cell. M_k represents the maximum polarization due to battery hysteresis as a function of battery SOC. The function represented by $F(i_k)$ is called the hysteresis model of the battery which represents the battery during dynamic environments. Hysteresis in battery occurs when the current changes from charging to discharging and the battery voltage profile does not follow the same trace. Equations 2.5, 2.6, and 2.7 are the state space representation of the lithium ion cell which include current and open circuit voltage of the battery as the primary inputs [44]. Using the state space representation mentioned above, SOC of the next time step is estimated. The initial SOC at time step k is assumed based on the battery parameters (for example, assuming the SOC as zero for a very low voltage). Kalman filter technique is mathematically exhaustive and requires a whole chapter on the analysis of it alone. The author suggests following [44] to dive into detail in the use of kalman filter for SOC estimation.

The advantage of using Kalman filter is that it can accurately predict states despite the ef-

fect of noise and external disturbances (the noises and external disturbances are considered Gaussian in nature). However, it cannot be used for non-linear systems directly and require significant assumptions. And it is mathematically intensive to perform. Apart from these methods there are other variations of Kalman Filter Algorithms that have been researched significantly for estimating battery dynamics and SOC. Extended Kalman Filter (EKF) [47] [48], Unscented Kalman Filter (UKF) [27], Sigma Point Kalman Filter (SPKF) [28] are some of the few algorithms that have gained wide research interests recently.

Besides these, there are other algorithms like Particle Filter [21], H∞ Filter [7], Recursive Least Square (RLS) [32], etc that have been used in estimation of Li-ion batteries state of charge.

All the techniques mentioned above rely on accurate models of Li-ion batteries and hence are termed as model-based estimations. Their major limitations is the obvious disadvantage of needing very accurate model of battery chemistry and battery dynamics. This can be a challenging task considering the battery dynamics can vary with change in temperature, applications, charging/discharging cycles and change in the electrochemical properties of the battery. However, with extensive database of battery cycling and charge/discharge parameters, use of learning algorithms like neural networks will help in abandoning the model based approach and with multiple training epochs, obtain acceptable accuracy. The remaining of this thesis deals with the use of neural networks for estimating the SOC of Li-ion cells.

2.4 Neural Networks

The field of neural network has been very popular recently due to their capability to find pattern and information from complex sets of data. Primarily developed to investigate the nature of logic gates, neural network have been used for wide range of applications like image processing, pattern recognition, data visualization, automation, etc. The reason it has been considered for estimating the SOC of Li-ion battery is because neural networks are robust and perform well as an alternative modelling technique for non-linear phenomenons. And as the electric vehicle industry and stationary storage systems comprising of batteries are on the rise, there are thousands of cycling data, performance indices parameters, charge/discharge data online. Using these data, the designed neural networks can be trained and tested to validate their accuracy and use in predicting the state of charge of the batteries. Artificial Neural Network (ANN) can be an alternative to traditional Multiple Input Multiple Output (MIMO) systems. This is because Artificial Neural Networks (ANNs) can outperform traditional Multi-Input-Multi-Output (MIMO) systems due to several key advantages:

1. Non-linear Relationship Handling: ANNs can model complex, non-linear relation-

ships between multiple inputs and outputs more effectively than traditional linear MIMO systems. This capability is crucial in applications where interactions between variables are not straightforward.

2. Learning from Data and Adaptability: ANNs excel at learning patterns directly from large datasets. They can automatically adjust their parameters through training to capture the underlying relationships in the data without needing explicit programming of these relationships. ANNs can adapt to new data and changing conditions through retraining. This adaptability is beneficial in dynamic environments where the relationships between inputs and outputs may evolve over time.
3. Scalability and Flexibility: ANNs can be scaled to handle very large numbers of inputs and outputs by adjusting their architecture (e.g., the number of layers and neurons). This flexibility allows ANNs to be tailored to a wide range of problems, from simple regression tasks to complex image and speech recognition tasks.
4. Feature Extraction and Generalization: ANNs, especially deep neural networks (DNNs) and Convolutional Neural Networks (CNNs), can automatically extract relevant features from raw input data. This automatic feature extraction reduces the need for manual feature engineering and can lead to better performance by capturing complex patterns that might be missed by traditional methods. Also, properly trained ANNs can generalize well to unseen data, meaning they can make accurate predictions on new, previously unobserved inputs. This generalization ability is critical for reliable performance in real-world applications.
5. Parallel Processing Capabilities: ANNs are well-suited for parallel processing on modern hardware, such as GPUs. This capability allows them to handle large datasets and complex models more efficiently, resulting in faster training and inference times compared to traditional MIMO systems.
6. Robustness to Noise and Variability: ANNs are generally robust to noise in the input data. During training, they learn to identify and emphasize relevant patterns while de-emphasizing random noise. This robustness makes ANNs suitable for applications where data quality may vary.
7. Integration of Diverse Data Types: ANNs can integrate and process diverse types of data (e.g., numerical, categorical, images, and text) within the same model. This integration is particularly useful in complex systems where inputs come from multiple sources and modalities.

While ANNs have these advantages, they also come with challenges such as the need for large amounts of labeled training data, significant computational resources, and careful tuning of hyperparameters. However, their ability to learn complex patterns and adapt to diverse problems often makes them superior to traditional MIMO systems in many applications.

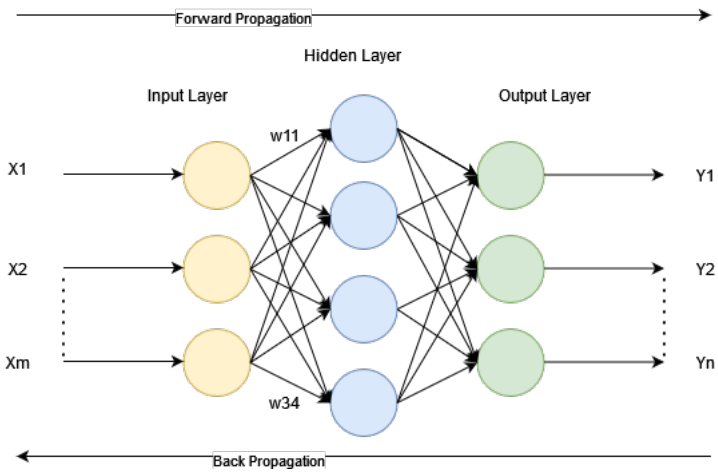

Figure 2.8. *Typical representation of a neural network showing an input, an output and one hidden layer.*

There are multiple books and resources on the working of neural network. [4] [2] [24] provides some excellent insights into the theory and design of NN. In summary, neural network use numbers as weight and biases to mimic the working of a biological neuron. It has an associated activation function that is used for decision making. Activation functions are also the main part that handle the non-linearity of the data. Figure 2.8 represents a simple neural network with three layers i.e. input, hidden and output. The weights is often represented as w and the bias in each neuron is represented with θ. The activation function gets input as the sum of weight and biases and provides an output based on the threshold set on it. It is the main decision making unit in the neural network. The bias is also added to take into consideration any input offsets. Mathematically in any nodes,

$$net_k = \sum_{1}^{n} w_{jk} H_j + \theta_{jk} \tag{2.8}$$

where H_j is the input to the node. And, the output of the node will be:

$$Y_k = F_k(net_k) \tag{2.9}$$

where F_k denotes any activation function being used. The most common activation functions are the sigmoid and hyperbolic tangent functions. Besides, sigmoid and tanh, there are multitudes of other activation functions that are used in neural networks like Rectified Linear Unit(ReLU), Binary step, Linear, Swish, Softmax, etc [54] [18]. RELU and its variations like Leaky ReLU (LReLU), Parametric ReLU (PReLU), Randomized ReLU (RReLU), Exponential Linear Unit (ELU) etc. are widely popular in the modern literature for their performance. The RELU function as shown in figure 2.9 is superior to other activation functions because of the ease of differentiation. The only drawback it has is that it is non-differentiable at $x = 0$. However, this problem can be solved by assumption, like the derivative at $x = 0$ can be assumed to be 0. This flexibility makes ReLU and its

different derivatives suitable for wide range of applications in machine learning research.

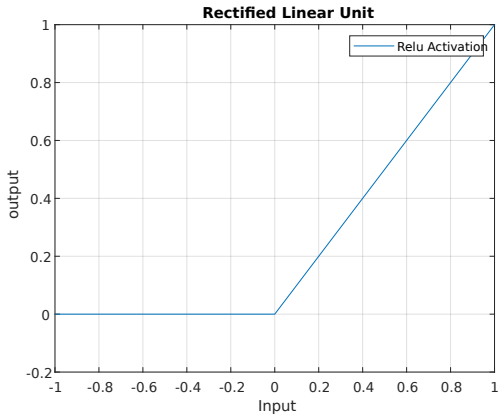

Figure 2.9. *ReLU activation function*

The decision making capacity of neural network is developed using a predefined set of data and labels wherein the network model tries to predict the label based on the corresponding data and minimize the cost function. This process is termed as "training of the model". When the model trains, it adjusts it weight and biases in such a way that the cost function is minimized. Once, the cost function minimizes to an accepted value, the model can be used for prediction on real data using any computers. The cost functions can be a simple mean squared error (MSE) or more complicated like cross-entropy errors for classification problems [52]. The error value is given as:

$$E_{ij} = Y_{predicted} - Y_{actual} \tag{2.10}$$

The Mean Square Error (MSE) is given by:

$$\xi_{MSE} = \frac{1}{N}\sum_{1}^{N}(Y_i' - Y_i)^2 \tag{2.11}$$

where Y_i represents the actual output and Y_i' represents the predicted output. The MSE mentioned in equation 2.11 can be the cost function and the main task of back propagation algorithm is to adjust weight and biases such that it is minimized. This is obtained as shown in equation 2.12 and 2.13. The new value of weight and biases are updated from the old values using the following equations:

$$w_{new} = w_{old} + \alpha\frac{\partial\xi}{\partial w} \tag{2.12}$$

$$\theta_{new} = \theta_{old} + \alpha\frac{\partial\xi}{\partial\theta} \tag{2.13}$$

where α is the learning rate: a constant that is used to speed up the "learning process". Initially, the value of w_{old} and b_{old} is selected randomly. With each iteration they are updated such that the final value will be the one that minimizes the cost function.

The partial derivative of the cost function as shown in the equation 2.12 and 2.13 is termed the gradient descent. In simple words, the gradient descent tries to find the minimum of the cost function. The neural network trains for multiple epochs (iterations) until the minima of the cost function is obtained. However,it need not be necessary for the gradient descent to find the global minimum. Hence, usually the algorithm converges to one of the many local minima in the cost function. The local minima is usually determined by the initial seed values (weights and biases). For different seeds, the local minima will be at a slightly different location. Once the machine error stabilizes to a certain steady value or reaches a pre-defined threshold, it has 'learned' and can be used to make predictions using the test values.

Apart from the MSE, there are other error metrics like RMSE (Root Mean Square Error) and Mean Absolute Error(MAE) used for analysing the performance of the neural network. The RMSE and MAE are defined as equation 2.14 and equation 2.15 respectively where the symbols carry the same meaning as in equation 2.11

$$\xi_{RMSE} = \sqrt{\frac{1}{N}\sum_{1}^{N}(Y_i' - Y_i)^2} \tag{2.14}$$

$$\xi_{MAE} = \frac{1}{N}\sum_{1}^{N}|Y_i' - Y_i| \tag{2.15}$$

This thesis uses two ANN of different nature to try and estimate the SOC of the Li-ion battery. Firstly, a feed-forward Deep Neural Network with two hidden layers was used to estimate the SOC of the Li-ion battery using an online database of experimental data. Secondly, a Long Short Term Memory (LSTM) Network has been used to estimate the SOC using the same dataset. While the detailed analysis of the experiments are presented in next chapters, a short introduction to these networks have been provided in the following paragraphs.

2.4.1 Deep Neural Network

A Deep Neural Network (DNN) (also called Mulitlayer Perceptron) is a type of feed-forward ANN architecture with two or more hidden layers between the input and output layer [23]. It is one of the two broad classes of neural networks, the other being Recurrent Neural Network (RNN). These networks are called feed-forward networks because the neurons

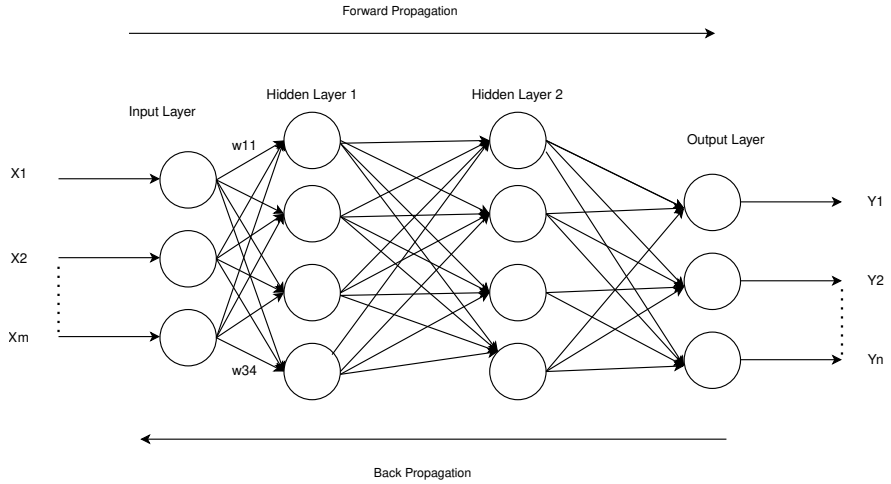

Figure 2.10. *A fully connected FeedForward Deep Neural Network (FFDNN)*

in this networks don't have any feedback loops. Hence, the flow of information is unidirectional, always from the input layer to the hidden layer and then to the output layer. Figure 2.10 shows a diagrammatic representation of a fully connected feed forward DNN. Consider a function f^* defined as:

$$y = f^*(x) \tag{2.16}$$

which maps any input x to a corresponding output y. The DNN attempts to approximate this function by defining a mapping function

$$y = f(x, \theta) \tag{2.17}$$

and learns the values of parameter θ that results in the best approximation of $f^*(x)$. A typical DNN has multiple functions f^a, f^b, f^c connected in chain like pattern to provide a final output defined as $f = f^a(f^b(f^c(x)))$[23]. The terminology 'Deep' refers to the multiple layers in this chain that provide *depth* to the neural network. Having more layers tend to increase the accuracy of the model at the cost of training time. Similarly, too many training epochs and the network cannot generalize which makes it unsuitable for dataset different than the training data. The number of hidden layers and the type of activation functions used in the DNN are dependent on the type of application. Aurelien Geron provides a detailed information on the design of DNN [22].

However, in cases of DNN with large number of hidden layers, the back-propagation algorithm suffers from the problem of vanishing/exploding gradient. This happens because the chain rule of differentiation that is used to compute the gradient descent has to calculate many gradients along the path to reach the input layer. When the output of the gradients from such functions either become very small or very large, they get multiplied over and over again, depending on the number of hidden layers, causing the gradient to

either vanish or explode. In such cases, the model doesn't train at all with the training data. Hu et al. [33] have mentioned a variety of options to mitigate this problem using different fail-check methods like the use of Gated Neural Networks (GNN) that introduce gates to control the information flow through the network or the use of Long Short Term Memory Networks (LSTM), etc.

2.4.2 Long Short Term Memory Network

Long short term Memory (LSTM) are a type of Recurrent Neural Networks (RNN) developed by Hochreiter & Schmidhuber. These are a different class of neural network compared to the DNN discussed before. A RNN is a family of ANN that is used for the sequential data processing. RNN uses the output of the hidden state from the previous time step as the input to the current time step. This feedback from the previous time-step hidden state to the current time-step hidden state, is the feature of RNN that makes it suitable for time series analysis. Figure 2.11 shows a typical RNN shown unfolded in time. The hidden layer has a feedback loop from the previous time step as shown from the small loop *h*. At each time step *t*, the input to hidden layers *h(t)* are the model inputs *f(t)* and feedback from previous time step hidden layer *h(t-1)*. The output for each time step is the state of charge of the cell represented as *SOC(t)*. The time steps *(t-1)* and *(t+1)* represent the previous time step and next time step respectively. The feedback is from the previous state hidden loop to present state hidden loop as shown in the figure 2.11.

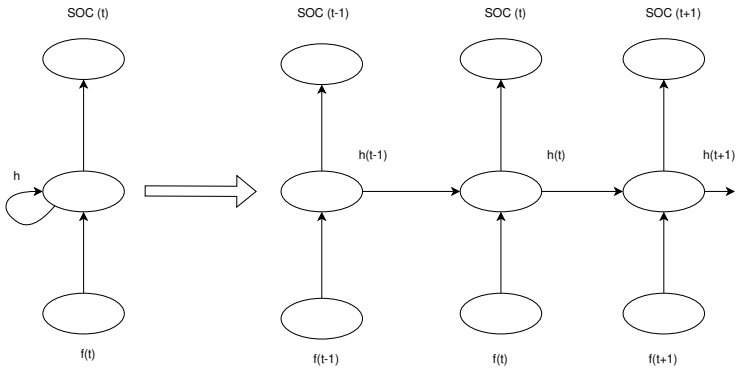

Figure 2.11. *A RNN architecture(left) and the RNN architecture unfolded in time*

However, a traditional RNN is only capable of handling one hidden state at a time. This makes it difficult to use where long term dependencies of data is necessary. For multiple hidden states, the neural network has to be unrolled for multiple steps in time. This creates difficulty in training the model. The more the unrolling of the model, the farther the gradient-descent for back-propagation has to travel. Since, the back-propagation algorithm exploits the chain rule of differentiation for updating the weights and biases, multiple unrolling causes this smaller values between 0 and 1 to be multiplied and accu-

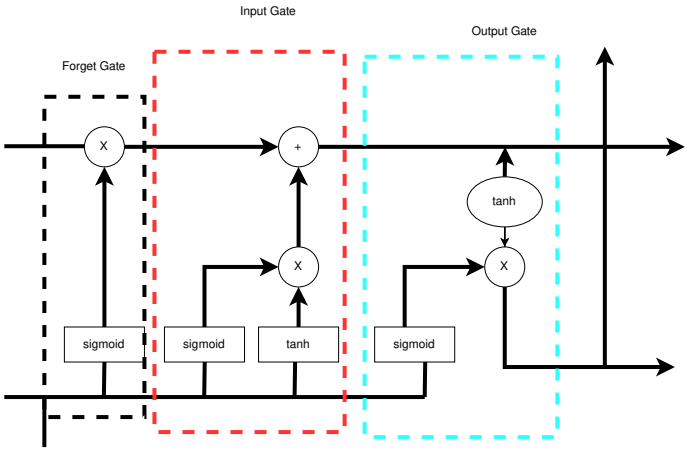

Figure 2.12. *LSTM architecture showing different gates*

mulate causing the gradients to vanish. Similarly, if the output of the activation function is greater than 1, this multiple unrolling can cause the gradients to accumulate and increase exponentially causing the gradients to approach infinity. which is termed as gradient explode. This problems is called the vanishing/exploding gradient problem. Hochreiter & Schmidhuber have tackled this issue in their seminal paper introducing the concept of Long short-term Memory (LSTM) network [29]. They introduced the forget gate to avoid the vanishing/exploding gradient problem. The forget gate only takes into account the information necessary for the model and ignores the rest of the information. The architecture of the LSTM network is shown in the figure 2.12 The input gate takes in account those values that are needed in the LSTM cells from previous time steps. They use a combination of sigmoid and tanh function to record only necessary data for LSTM cell. The forget gate gets rid of values that are not necessary for the LSTM cell. The output gate finally extracts the needed output from current time step for the next time step.

The following literature provide explanations about LSTM in great detail. [8] [29] [64].

LSTM networks process data sequentially. If the training data consist of thousands of data points, which is usually the case, feeding all data at once will take too much memory as all these data are stored in memory before being fed to the model. For example, consider in this case, where there are 2 million input data points. Feeding 2 million data at once to the model will clog the memory. This will drastically reduce the training time and performance of the model. Hence, in LSTM networks concept of batch is introduced. The input sequential data is divided into multiple small batches and then fed to the network one batch at a time. The size of the batch depends on the memory capacity of the machine and the number of data points used to train the model.

Chemali et al. [8] have proposed the use of LSTM network for estimation of state of charge of the Li-ion battery. Following on their footsteps, the author has developed a simplistic model of LSTM network for estimation of SOC using the same dataset as used

in FFDNN described earlier.

2.5 Literature Review

The Back Propagation Neural Network (BPNN) is one of many neural network models that have been used for the prediction of SOC of Li-ion batteries. Cui et al. [17] have provided a comprehensive list of neural networks including Time Delay NN, Elman NN, Radial Basis Function NN, Convolutional NN, Nonlinear Auto-regressive with Exogenous Input NN (NARXNN), Gated Recurrent Unit (GRU), Long Short Term Memory Network (LSTM) that have been used to predict the SOC of Li-ion batteries.

Wang et al. [67] used BPNN with Artificial Fish Swarm Algorithm (AFSA) to predic the SOC of power Li-ion battery. The battery used in this experiment was a 48 V 50 Ah $LiFePO_4$ power battery. This paper used the AFSA algorithm to optimize the BPNN. AFSA algorithm is a stochastic search optimisation algorithm which is based on simulating the behaviour of fish foraging, clustering and tailing. The absolute error using this process was less than 0.5%. The parameters considered for the estimation of SOC are the batteries voltage, current and temperature data. Three layers i.e. input, hidden and output layer were used in this case . The paper concludes that at low SOC values, the AFSA-BP algorithm can estimate the SOC value more stably and accurately than when the weights and biases are initialized randomly. This paper mentions that true value of SOC is not available because of the difficulty in measuring the SOC. The calculation of error is obtained by agreeing the true value to be the one obtained from theoretical calculations or by estimating using more sophisticated instruments. Also, in-terms of time complexity, this model is not much faster than Extended Kalman Filter (EKF).

Traditionally Support Vector Machines (SVM) [45], Fuzzy Controller [53], Artificial Neural Networks (ANN) [10] and a combination of other algorithms were used to predict the state of charge of the battery. These model employed only two computational layers due to hardware and software constraints. But with the increase in computational capacity of modern computers and state-of-the-art hardware and software systems, there can be multiple computational layers working together. This augmented version of ANN with multiple computational layers is termed as Deep Neural Network (DNN).

The use of DNN to estimate the state of charge of Li-ion battery is fairly new[30]. How et al. [30] have used DNN to predict the SOC of Li-ion batteries achieving a Root Mean Square Error (RMSE) of 3.68%. They found out that using multiple hidden layers (up to four hidden layers) decreases the error rate and improves the SOC estimation. However, the additional increase of the hidden layer negatively impacts the model performance. They have used TensorFlow and Keras deep learning framework with two NVIDIA 1080Ti graphical processing units to train all the DNN models. They used a cylindrical cell INR 1865020R $LiNiMnCoO_2$/NMC Li-ion battery for three different sets of temperatures (0 °C,

25 °Cand 45 °C) to gather the battery data. The source of their data is from the Center for Advanced Life Cycle Engineering Battery Research Group [69]. The instantaneous voltage and current data while discharging was sampled at a frequency of 1Hz. The experiment has been run for 5000 seconds (approx. 1.4 hrs) only. Thus, it is safe to say that they have not undertaken the full life time of the cell. With 3 hidden layers and a total parameters of 9,409, this model was able to achieve a RMSE of 3.68%. The ground truth model for the model is obtained by estimating the SOC using Coulomb counting in this case as well. The author states that this proposed method was better than OCV, CC, UKF, LSTM, CNN-GRU with the lowest RMSE and MSE. While the author states that the database is an open database, there is no mention of discharge C-rate and aging of the cell. The authors also agree that while DNN performed well, in terms of error metrics, Recurrent Neural Network (RNN) perform better than DNN. The hyper-parameters are selected empirically and no mention of how the model behaved for different starting conditions have been given. The authors have used single drive cycles (i.e. drive cycles are not combined) which is not representative of real life EV driving. They have selected the number of layer based on the Mean Square Error(MSE). The ground truth for the model is obtained by performing the Coloumb counting approach and the Q_{usable} as mentioned in equation 2.1 has been replaced by the nominal capacity of the battery. This indicates that they have not considered the effect of aging in battery SOC estimation using NN.

Chemali et al. [9] have proposed the use of feedforward DNN to estimate the SOC of batteries where the battery measurements are directly mapped to battery SOC. The proposed model uses 2 layers and 256 neurons and has a Mean Absolute Error (MAE) of 1.10% with a maximum error of 2.38% at 25°C. The training data was generated in a lab in this case by applying drive cycle loads at various ambient temperatures to a Li-ion battery to simulate real-world dynamics. To strengthen the DNN, the training data was augmented by injecting Gaussian noise (specifically, a normally distributed random number generator has been used to generate noise for each of the battery signals). The input vectors that were fed to the network is given by $\psi(t) = [V(t), T(t), I_{avg}(t), V_{avg}(t)]$ where V(t), T(t), $I_{avg}(t)$ and $V_{avg}(t)$ represent voltage, temperature, the average current and average voltage of battery at a given time 't'. The average value of voltage and current was calculated over different precedent time-steps ranging from 50 to 400 time-steps. The author states that average values have been used instead of new input parameters as it improves computational efficiency and has low memory requirements. The author compared the computational speed of the designed neural network with an Extended Kalman Filter (EKF). It was found that NN took 0.07 seconds to complete the computation, while EKF took 0.66 seconds. The computation was carried out on a 13000 data points to estimate the SOC. It was observed that DNN was an order of magnitude faster than EKF. They performed this test for 160,000 epochs and tested for different number of layers. While the MAE decreased when the number of hidden layers was increased from

the initial 2 to 4, and 6, further increasing the hidden layers actually had no effect in the MAE due to the vanishing gradient problem. The authors also suggested that the model can accurately estimate the SOC even when there is 5 to 10°Cchange in temperature. However, the author doesn't mention what discharge C-rate has been used in the experiment. Similarly, the experiments have been only conducted for 7000 seconds (i.e. approx 2 hours). This doesn't provide enough information for considering the aging of the cell. Similarly, they have used each drive cycles independently. However, batteries in EV undergo different drive cycles at a given time. So, the experiment may not be representative of the real life scenario.

On another paper, Chemali et al. [8] have proposed the use of Long-Short Term Memory Network- Recurrent Neural Network (LSTM-RNN) to estimate the state of charge of the Li-ion battery. This model was able to achieve a MAE of 0.6% with a maximum error of 2.6% at 25 °C. Tensorflow along with two GPUs: NVIDIA Titan X Pascal and GeForce GTX 1080 TI were used to speed up the training process in the model. The input vector in this case was $\psi(t) = [V(t), T(t), I(t)]$ measured at each time step 't'. The cell they conducted these tests on was a panasonic 18650 battery cell with a $LiNiCoAlO_2$ or NCA chemistry which was also used in some Tesla vehicles. To generate the battery data, it was exposed to a selection of 10 drive cycles on different ambient temperature ranging from 0°Cto 25 °C. The training case data was selected from these 10 drive cycles on three ambient temperatures (0°C, 10°Cand 25°C). The whole of the testing was performed at the Wisconsin Energy Institute at the University of Wisconsin-Madison. This paper concluded that LSTM-RNN was able to offer competitive estimation performance when compared to other algorithms. One important observation they made was that the battery heated significantly, up to 13 °Cdue to higher loss at lower temperature. However, there is no mention of aging of the cell during the experimenting process. The authors also do not mention why only voltage, current and temperature have been selected as the inputs. There is mention of the models ability to encode weights and biases for different ambient temperature but it is unclear if the model works outside of the temperatures it has been trained on. Due to long chain of input sequence the input have been divided into different batches and the model has been fed with one batch at a time. However, the paper is unclear about how big the batch is or how should the batched be defined. Also, there is no mention on how the model behaves if the batch size is altered. The data for the model is generated by cycling a 35kWh cell and scaling the power profile to match a single cell. However, instead of scaling the model for a single cell, experimenting on a single cell might be more effective as the data generated is true and pure in a sense that it doesn't need scaling. The C-rate for discharge is 6C (about 18 Amps) for the battery. But batteries tend to discharge at different C-rates based on different road condition. The model however, hasn't considered these options perhaps due to difficulty in obtaining data for these scenarios.

Chaoui et al. [6] used a Time Delay Neural Network (TDNN) to estimate the SOC of $LiFePo_4$ cells achieving a RMSE of 0.0027 while testing after 650h milestone. A $LiFePO_4$ battery is subjected to a predefined calender ageing process using the experimental set-up. A supervised multi-layer perceptron (MLP) network was used to predict the ageing and estimate the SOC of the battery. This scheme uses unit time delays (UTD) to hold data corresponding to battery voltage at past time frames. Namely, ten UTDs are used for voltage learning which implies that battery voltage measured at each second for the past ten seconds is used as an input for the MLP. No UTD was used for current measurement. Hence, this network worked on thirteen inputs with batteries present voltage , current, temperature and the voltage measurement from the past ten seconds. The input vector was normalized in this case such that each value ranges from (0,1). The paper concludes that despite the effects of hysteresis, flat voltage profile, temperature and aging, the technique estimates the required quantities with minimal error. However, in this paper, the batteries are discharged without any drive cycles. The battery under test is just randomly charge and discharged. Also, cell aging and parameter selection has not been mentioned. The paper mentions using only 2 hidden layers with 3 neurons in first layer and 2 neurons in second layer. But there is no mention of training time and total parameters. It has also been mentioned that the model uses 10 UTD inputs from voltage value but not from current. The reason for this has not been clarified. Battery current is an important parameter for SOC estimation. The 13 inputs to the model include the present time step voltage, previous 10 UTD voltage, present time step current and present time step temperature. However, NN are computationally intensive in themselves. Using Unit Time Delays (UTD's) will further reduce the computational speed. There is no mention of how long did the training process lasted. Also, instead of using 13 inputs with just UTD voltage values, the previous papers have found much success with 3 inputs of voltage, current and temperature only.

Ren et al. [50] opted for Long Short Term Memory (LSTM) network based on Particle Swarm Optimization(PSO) (PSO-LSTM) for the estimation of SOC achieving a MAE (with noise) of 0.3493% and RMSE of 0.45%. This paper utilized the concept of particle swarm optimization to optimize the hyper-parameters of the LSTM network to match the data characteristics of Li-ion battery with the network topology. Also, to improve the anti-interference ability of the network, random noise is added to the input layer. While a normal LSTM method achieved a MAE of 0.621%, PSO-LSTM was able to achieve a MAE of 0.3493%. This paper states that PSO overcomes the shortcomings of artificially determining LSTM parameters and reduces the workload by optimizing the number of hidden layer neurons, learning rate and iteration time of LSTM. This paper doesn't dive into the problem of SOC estimation itself, but rather focuses on using optimization algorithm for obtaining hyper-parameters value in the network that result in minimum error. While the error did reduce, the reduction is not significantly high. Besides, the errors are

well below acceptable limit even without the optimization. This paper however doesn't deal with the effect of different ambient temperature on battery performance. Also, there is no mention of aging when the battery is cycled for long charge/discharge cycles. The experiment has been performed at only one temperature case i.e.25°C. Possibility of generalization of the model for different temperature cases is not shown here. Also, in-order to train the model, the authors have proposed generating false data. Unless the falsely generated data corresponds to real life cases, the NN model isn't useful enough for real life BMS. Also, the range of parameters for number of hidden layer neurons, learning rate and maximum is provided. But, there is possibility of optimum values lying beyond this range. And, increasing the range won't be of much help as it would be computationally cumbersome and it's better to randomly initialize the hyper-parameters.

Huang et al. [34] used Convolutional Gated Recurrent unit-Recurrent Neural Network (CNN-GRU) for SOC estimation of Li-ion batteries achieving a MAE of 1.3% at 26 °C. This paper showed that the proposed method can achieve higher estimation accuracy than recurrent neural network, gated recurrent unit, support vector machine and extreme learning machine for SOC estimation. The BAK 18650 battery with Li(NiCoMn)O_2 chemistry was used for testing purpose. The input vector in this case was also voltage, current and temperature. The paper states that CNN-GRU can directly map the battery measurement signals to SOC without the use of complex models and algorithms like Kalman filters. The paper concludes that CNN-GRU can estimate SOC for different ambient temperatures outperforming other neural network models. However, to consider the effect of battery aging, the parameters are expected to be updated regularly. Besides, the model is using only one drive cycle which might create difficulty to generalize the result.This is one of the few papers to talk about aging and provide the approach to deal with it. The authors suggest updating the model after every 2-3 months. This is particularly effective as the model is retrained after certain interval of time to learn battery dynamics as it gets old. The true value of the SOC is obtained from current counting method in this case as well emphasizing that the current sensor is calibrated carefully and the battery is always discharged at 100% SOC. The authors also assume that a fully charge battery has 100% SOC which is not always the case. For instance, a fully charge battery may have only 80% usable charge as it gets old. Besides, the effect of temperature also reduces the capacity of the cell as low temperature causes the internal resistance of the battery to increase. This is one of the reason why this model has performed poorly for low temperature cases.

3. METHODOLOGY: EXPERIMENTAL SETUP AND NEURAL NETWORK DESIGN

Lithium ion batteries have been used for wide range of applications in modern times. Therefore, the charge/discharge characteristics displayed by the battery depend on the specific application in which it is utilized. For example, the discharge profile of the battery used in an electric vehicle is different from that of a hand-held electronics device.The primary focus of this thesis is on estimating the state of charge (SOC) within electric vehicle applications, thus emphasizing the data acquisition and analysis process specifically for this context. Thus, the experimental tests are designed in such a way that the battery experiences a charge/discharge profile akin to that of an electric vehicle (EV). This way it is assured that the data generated is representative of the intended application.

Theoretically, datasets with minimum noise and other irrelevant information is considered a good set for training of neural networks. However, due to measurement and human errors during the data acquisition, noise in a data sets are unavoidable. Different filtering methods and data preparation algorithms can be used to siphon out the noise and make the NN model robust. Up sampling is done to some measurement data for similar time step representation. A detailed explanation on data acquisition, the experimental setup used for generating the training and testing datasets, and the methods used for preparing the final working datasets is described in detail as follows.

The experimental setup with the equipment used to generate the charge/discharge pulse is also shown in the figure. 3.1. The basic idea of measuring the ideal state of charge (SOC) for a battery through experimentation involves repeatedly charging and discharging the battery and monitoring the current over the charge/discharge period using an ampere-hour (Ah) meter. This is usually performed within a controlled environment, where the temperature, pressure and other environmental parameters can be set manually. The device offering this facility is called a thermal chamber or a climate chamber. The charge/discharge cycle is repeated until the battery's end-of-life is reached. In the case of a typical cylindrical 18650 lithium ion cells, this can be up-to 300-500 cycles. The term '18650' is a battery terminology used by researchers and EV manufacturers to represent a cell where 18 means the diameter of the cell in millimeters and 650 is the height of the cell in millimeters. This test is also important in determining the SOH of the battery as

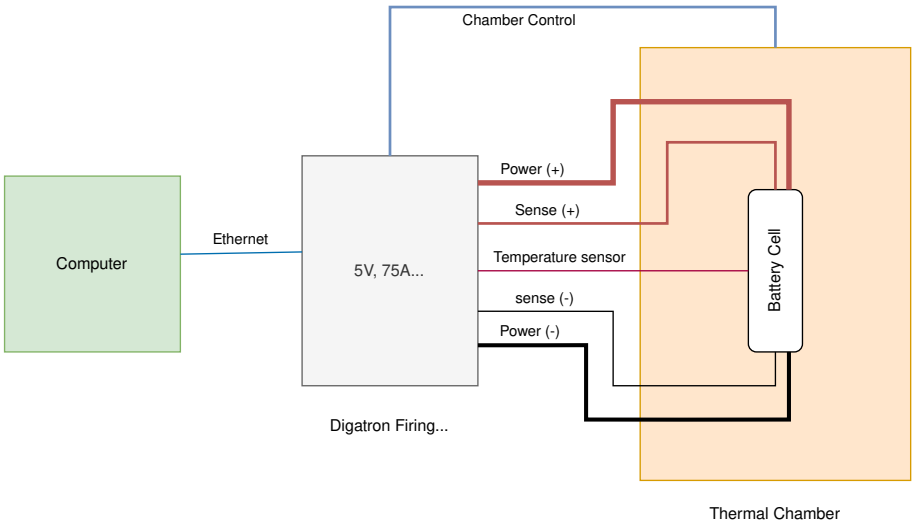

Figure 3.1. *Schematics of test setup and data logging system*

repeated charge/discharge cycle helps in analysing aging of the cells. The Ah-meter data is logged on a computer, which, also controls the battery cycling device. The Digatron Firing is one of the many pulse generators available in the market that is used to generate charging currents to charge the battery. Similarly, it has load resistors built into it for the batteries to discharge. The power cables are used to transmit charging and discharging current while the sense cables are used to measure the current entering and exiting the circuit. A cable for temperature sensor is also used to observe the temperature of the cell during the whole charging discharging cycle while the temperature of the chamber is provided by the chamber control signals. The whole data logging process is done in the computer that is connected to the pulse generator via an Ethernet cable or by any other means possible.

Although Coulomb counting is considered error-prone, it remains the simplest and easiest technique for estimating the SOC of a battery in the laboratory. Much of the literature on using ANNs for SOC estimation relies on Coulomb counting data as the true SOC value for their models. This approach has also been adopted in this thesis. While Coulomb counting can drift over the long run, this can be mitigated with carefully calibrated current sensors. The effect of aging on the Q_{usable} charge is ignored due to the lengthy time required to observe significant effects. Aging occurs slowly when temperature and usage conditions are not extreme. It is estimated that EV batteries degrade about 2-3% per year [3], which is minimal. Therefore, in this experiment where the battery is tested for a few hours in each temperature cases, battery aging is ignored.

The batteries were subjected to repeated charging and discharging and the charge entering/exiting the battery was measured using an Ah-meter. After each time-step, the measurement data consisting of voltage, current, temperature, was logged into the computer system. In order to make the training data representative of the EV application,

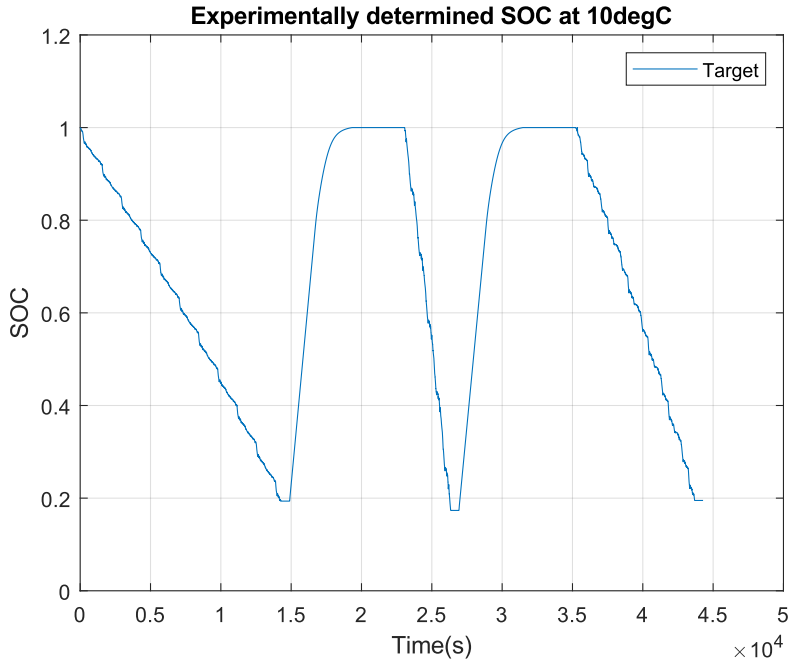

Figure 3.2. *Discharge pattern of the test battery subjected to different drive cycles*

the experiment was performed using a series of vehicle drive cycle discharging profile. A drive cycle is a curve representing the speed vs time of a typical vehicle, which is generally used to measure its emission and fuel economy. The drive cycle provides information like how and at what speed vehicles travel. This serves as a reference for discharging the test battery so that the discharge profile resembles the performance of the on-board battery units in an EV. A list of drive cycle profiles used for testing is provided by EPA [62]. The drive cycles used for the test in this experiment were the UDDS (Urban Dynamometer Driving Schedule),HWFET (Highway Fuel Economy Driving Schedule),LA92 and US06. The speed vs time curves for all these drive cycles can be found in United States Environmental Protection Agency (EPA) websites[62]. Figure 3.2 shows a combination of discharge profile used during the experiments.

The final model has both the predictors (voltage, current and temperature) and the corresponding labels (SOC). The data set was prepared by concatenating these predictors and labels into one matlab structure. The error data points and spikes were removed while the SOC was calculated by dividing the ampere-hour data by the cell nominal capacity (keeping the discharge start at 100% and the charge end at 100%). Each data file has been acquired at a sample frequency of 1Hz to maintain synchronicity.

The input data were normalized and scaled between 0 and 1 before feeding into the neural network model. For instance: instead of directly feeding the neural network with (4.2V, 15A, 35°C), the data points were normalized to (1,1,1) [65] using the z-score normalization given by equation 3.1.

$$x_{norm} = \frac{x - \mu}{\sigma} \tag{3.1}$$

where μ and σ represent the mean and the standard deviation of the data respectively and x_{norm} is the new normalized value. The data has then been re-scaled so that they lie between 0 and 1. This has been achieved by using the min-max scaling given by equation 3.2.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.2}$$

where x_{max} and x_{min} are the maximum and minimum value of the particular feature in the dataset respectively. Normalization and scaling of the data helps to increase the accuracy of the model by resizing the working dataset into smaller workable scale. For instance, the temperature of the cell ranges from 25°Cto 35°Cwhile the voltage of the cell ranges from 2.6 V when empty to 3.6 V when full. There is a wide difference in the scale of the data. Neural network models perform poorly if the scale of the data is different and the variation is large. This can be avoided by re-scaling all data to the same scale. For instance in this model, the data are re-scaled between 0 and 1.

Instances where the input is outside of this range will have effect on the performance of the model. Input data that are too large or too small compared to the scaled data will affect the output of the activation function causing unexpected behaviour and network's loss of generalization i.e. unable to perform on unseen data. Similarly, extreme data can affect the numerical stability of the network in deep networks with multiple layers. Besides, the network may extrapolate and produce biased output that doesn't make sense. Hence it is of utmost importance to prepare the data via normalization and scaling. Further explanation about this approach has been done in the next section.

The battery was charged using Constant Current- Constant Voltage (CC-CV) charging process. In this process, the battery is initially charged using a constant current (of 1C) until it reached a fully charged voltage. After that, the voltage is kept constant until the current reached the cut-off value. At this point the battery is assumed to be fully charged. Figure 3.3 shows the charging of the battery using CC-CV process during the 10th charging cycle at 25°C.

The test was performed on a LG 18650HG2 cell at McMaster University by Dr. Phillip Kollmeyer [46]. The test specifications are presented in table 3.1 while the battery specifications (taken from the data sheet) are presented in table 3.2. As per the test description,

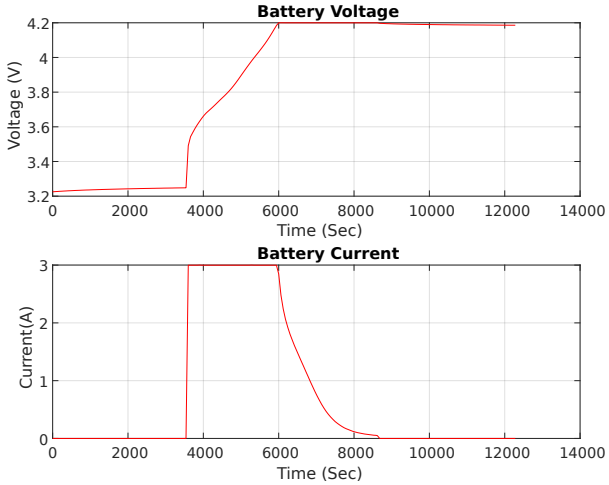

Figure 3.3. *CC-CV charging represented in the graph*

Battery	Capacity	Format	Profiles	Temperature
LG-HG2	3Ah	Cylindrical 18650	Mix(1-8), UDDS, LA92, US06, CC	-10°C, 0°C, 10°C, 25°C

Table 3.1. *LG-HG2 has been chosen as the test battery*

"*A brand new 3Ah LG-HG2 cell was tested in an 8 cu.ft. thermal chamber with a 75amp, 5 volt Digatron Firing Circuits Universal Battery Tester channel with a voltage and current accuracy of 0.1% of full scale*".

An important terminology in the battery charging/discharging process is a C-rate. A C-rate is the measure of battery current at which it is discharged relative to its maximum capacity. A 1C discharge rate means that the discharge current will discharge the entire battery in 1 hour. For instance, if a fully charged 10Ah cell is discharged at 1C rate, the discharge current is 10A and the battery is completely discharged in 1 hour. If this cell is discharged at 2C rate, the discharge current is 20A and the battery will be completely discharged after 30 minutes. Similarly, if the cell is discharged at 0.5C, the discharge current is 5A and the cell will be completely discharged in 2 hours.

Chemistry	$Li[NiMnCo]O_2$ / Graphite + SiO
Nominal Open Circuit Voltage	3.6V
Min/Max Voltage	2V/4.2 V
Mass/Energy Storage	240 (Wh/kg)
Nominal Capacity	3Ah
Discharge	2V (Cut off) Voltage, 20A Max Continuous Current

Table 3.2. *Battery specification from the data sheet*

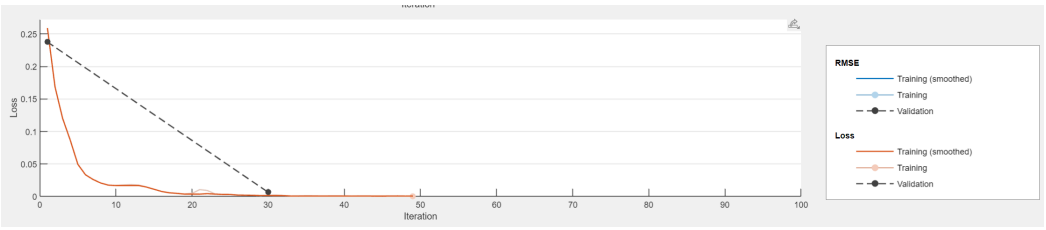

Figure 3.4. *Snapshot of model's training loss as iteration increases*

While the original experiment was conducted at six different temperature(40°C, 25°C, 10°C, 0°C, -10°C, and -20°C), for simplicity and ease on the machine (for training and testing), only four temperature measurements (25°C, 10°C, 0°C, -10°C) have been used for analysis in this thesis. Mendeley database [46] provides more information on the nature of experiment and also provides a live Matlab script to visualize the data and train the model. However, for the thesis, the author has implemented a neural network from scratch with the perimeters borrowed from Vidal et al. [65] and Chemali et al. [9]. The model of the NN coded in MATLAB is provided in Appendix A.

The model for the Deep Neural Network (DNN) has been coded in MATLAB using the neural network toolbox. The code was written in standard MATLAB. The data was organized into training ,validation and testing sets. Th distribution of these dataset is done as shown in the following table.

Items	Count	Percentage
Training data set	669,956	75%
Validation data set	39,293	5%
Testing data set	173,624	20%
Total	882,873	100%

Table 3.3. *Distribution of data into training, testing and validation sets*

The total data set for all temperature cases consisted of a total of 882,873 data points of voltage, current and temperature. These data are obtained from a new LG-HG2 cell. The cell is not an EV module in itself but a representation of the cells present in an EV module. Each cells in an EV module undergo discharge profile akin to that in figure 3.2 when an EV is driven in the real world. It can be noticed that To avoid over-fitting, the training and testing data were separated early on in the data preparation. The measured data points consisted of the voltage, the current and the temperature with the ground truth being the ideal SOC of the battery experimentally determined by measuring the current entering and exiting the battery

The model consists of an input layer, two hidden layers and an output layer. There are five number of inputs viz. voltage (V(t), current (I(t)), average voltage $(V_{avg}(t))$ average current $(I_{avg}(t))$ and temperature (T(t)) and one number of output (SOC(t)).Initially different

activation functions were experimented to contrast their performance and select the best one. The activation functions that were finally selected were the tangent function in the input layer, leaku ReLU in the first hidden layer and clipped ReLU in the second hidden layer followed by a regression layer based on a model designed by Vidal et al.[65]. The matlab neural network toolbox provides functions *trainingOptions()* and *trainNetwork()* to build and train the neural network. Another function, *predict()* was then used to make estimations of SOC using the designed network. The performance of the model was observed through the training progress graph as shown in figure 3.4. The graph shows the training loss calculated at each iteration plotted against the corresponding iteration value. The decrease in the training loss indicates that the model is learning the data as the iteration progresses. Every 30 iteration, the model is validated with data set from the validation set. This was also decreasing as the training progressed, suggesting that the model is able to generalize on new data too. It is to be understood that the validation data set is different from the training data set which is used to check if the model is able to generalize on new unseen data or not. The less the validation loss, the better the model is at generalizing on new data.

While the *tanh* function was chosen as the activation function, Vidal et al. [65] used "Rectified Linear Unit" (Relu) functions and it's variation for the hidden layers showing remarkable increase in performance. Hence, two versions of Relu functions have been used as non-linear activation function in the hidden layers. They are the leakyRELU and clippedRELU. Figure 3.5 and 3.6 show the graph of the leaky ReLU and Clipped ReLU respectively. The clipped ReLU is clipped at 1 as SOC greater than 1 is not logical. So the ceiling is set to 1.

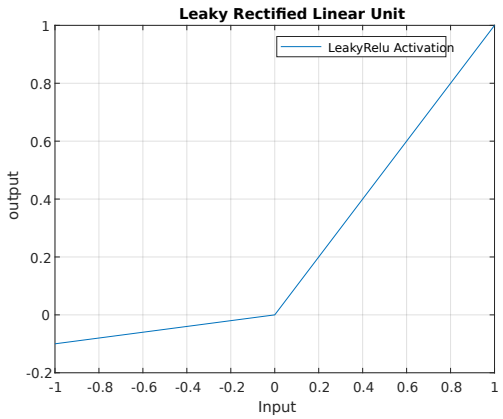

Figure 3.5. *leakyrelu activation function*

The list of parameters used for building the model are presented in table 3.4.

After the data collection and preparation was complete, training of the neural network was started. The model was run for 1000 epochs (iterations) using the parameters specified in table 3.4. A Lenovo Legion 5 Pro Ryzen 7 laptop with a NVIDIA GeForce RTX 3070 GPU

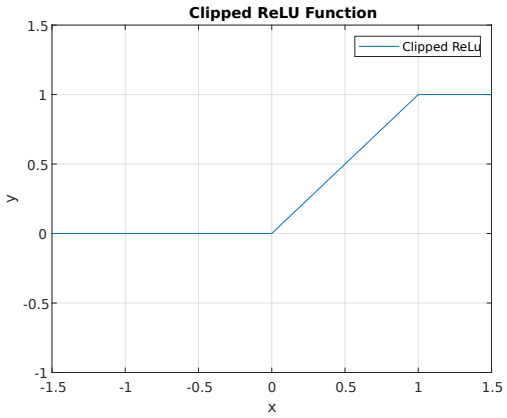

Figure 3.6. *clippedrelu activation function*

Parameters	Values
Number of Inputs	5 (Voltage, Current, Temperature, Average Voltage, Average Current)
Number of Outputs	1
Epochs	1000
Mini-batch-size	1
Initial Learning-Rate	0.01
Learning Rate Drop Period	200
Validation Frequency	30 iteration
Number of neurons per hidden layer	67
Input Layer	Sequential Input
First Hidden Layer	$\tanh$ activation function
Second Hidden Layer	Leaky ReLU(0.3)
Output Layer	Clipped ReLU (1)
Optimizer	Adam (Adaptive Moment Estimation)

Table 3.4. *Initial parameters used for building the NN model*

was used as the training and testing machine. The option to select the GPU is provided in the MALTAB's *trainingOptions()* function. A validation set was used to "validate" the training process and fine tune the hyper-parameters. The data in validation data set were unseen data and hence helped in avoiding over-fitting and under-fitting of the curve over the training period. This approach helps to generalize the model for different test cases.

In-order to find the optimum number of neurons in the hidden layer brute force approach was used. Many literature provide different optimization approaches to find the parameters of the NN [35] [12] [49]. However,due to the simplicity of the brute force approach, it has been used to find the optimum number of neurons that give the lowest mean absolute error. Figure 3.7 shows the plot of different error cases when the number of neurons per

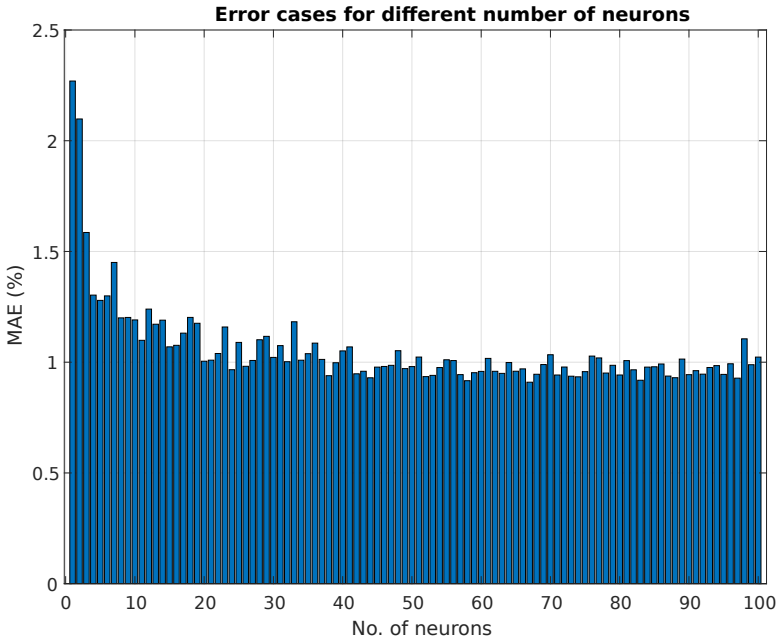

Figure 3.7. *Error cases for different number of neurons in the hidden layer*

hidden layer is changed. The low error cases were found when the number of hidden layer was 58, 67 and 83 with 67 neurons giving the lowest mean absolute error. Thus, the number of neurons was selected to be 67.

After the model was trained, it was tested against the test dataset and the errors were calculated. Three error metrics namely, MAE, RMSE, and maximum error have been plotted for visualization. The test data were grouped into four categories based on the temperature that they were measured i.e. -10°C, 0°C, 10°C, and 25°C. Figure 3.8 shows the SOC's categorised based on the temperature of the battery.

As mentioned in the above section, apart from the three measured parameters (voltage, current and temperature) there were two extra parameters that have been used to train the model and make predictions. Those were the average voltage and average current. This was done to improve the performance of the NN model. One major issue that was faced with the initial three input model (without the average voltage and current) was the model's poor performance. One of the reasons of such poor performance was that the time series data had no information regarding the previous inputs. And since FFDNN doesn't have feedback like RNN, they are at a disadvantage when processing time series data. This average of the voltage and current values acts as a memory of the previous inputs and hence helps in improving the performance of the model. Vidal et al.[65] suggested using the average of voltage and currents from previous time steps as additional input parameters for this reason. These parameters were computed based on 500 previous one second time steps measurements [65] where 500 was an chosen arbitrarily. After using the average voltage and current as the additional input parameters, the er-

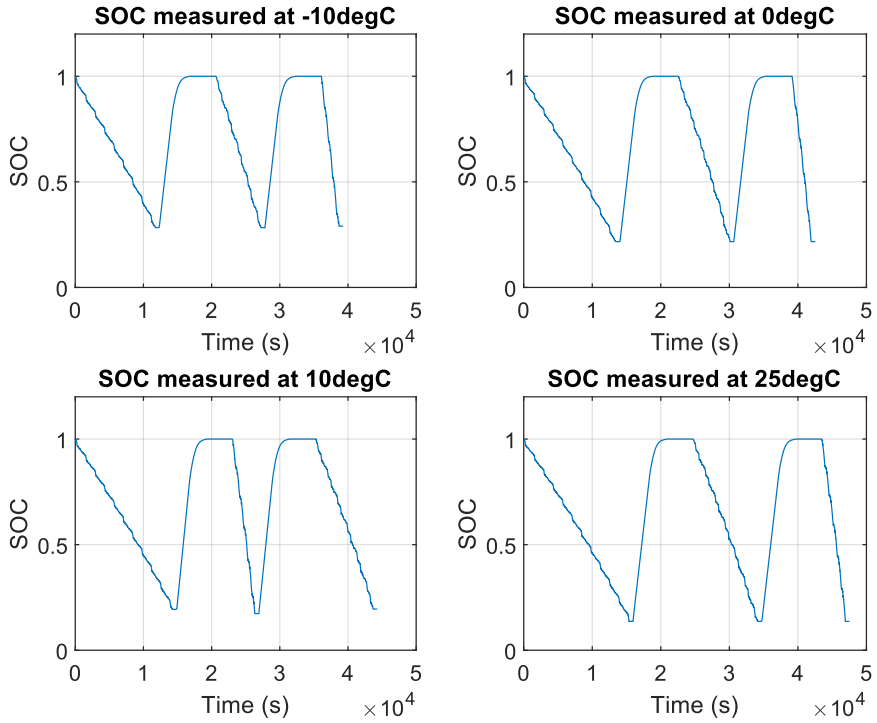

Figure 3.8. *Experimentally determined SOC which is the reference for the test cases*

rors decreased remarkably, thus improving the model performance. Figure 3.9 shows the SOC estimation at 25°Cusing only the three inputs and figure 3.10 shows the same estimation done using five inputs. Section 4 delves further into the comparison between these models.

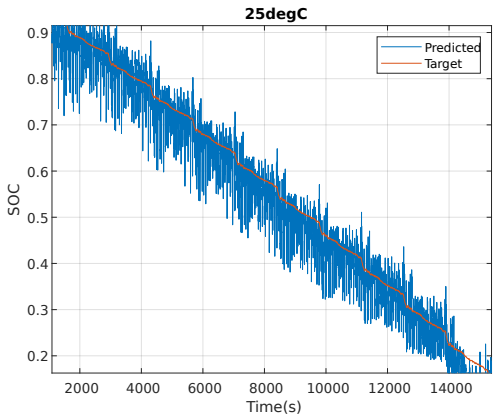

Figure 3.9. *SOC estimation with only three inputs*

The DNN used in the above case is an example of Feed Forward Deep Neural Networks (FFDNN) where the information flow is only in the forward direction. One issue with these kind of NN is that there is no information about the previous inputs or "memory" of previous inputs. This makes the NN inefficient for time series data where previous state could prove crucial for predicting the future states. Based on these short comings, Chemali

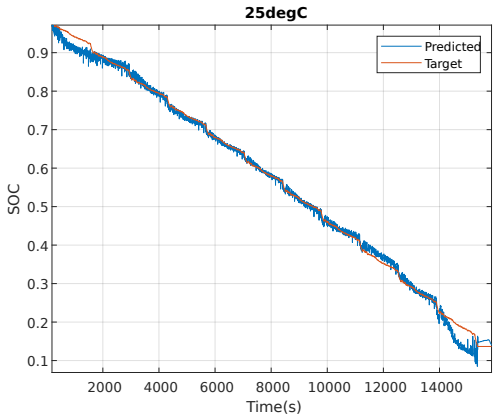

Figure 3.10. *SOC estimations with five inputs*

et al. [8] proposed the use of Long-Short Term Memory Network (LSTM) for predicting the SOC of the lithium-ion batteries. LSTM are a type of Recurrent Neural Networks (RNN), the other type of Artificial Neural Networks. Unlike FFNN, RNN have the ability to store previous time step information and are generally used for pattern recognition in sequential datasets [8]. While classical RNN suffers from the vanishing gradient and exploding gradient problems in long dependencies, RNN with LSTM cells seems to have had better success at capturing long-term dependencies [8]. Hence, along with the DNN, another model consisting of LSTM layer was also introduced to analyze the estimation capability. The parameters used to model the LSTM network are mentioned in table.

Compared to DNN, training a LSTM network takes longer time because of the feedback. Hence, unlike 1000 or more epochs that was used to train the DNN, LSTM network was trained just for 200 epochs, although the performance was comparable to DNN (results section deals with this in more detail). The specification of the machine used for training is the same as mentioned in the previous section. Table 3.5 mentions the parameters used in the design of LSTM cells.

Parameters	Values
Number of Inputs	3 (Voltage, Current, Temperature)
Number of Outputs	1
Epochs	200
Mini-batch-size	64
Initial Learning-Rate	0.01
Learning Rate Drop Period	30
Validation Frequency	50
Number of hidden neurons	256
Input Layer	Sequential Input

Table 3.5. *Parameters used in LSTM network design*

As the LSTM network have the feedback loop, they do not require the average voltage and average current parameters as additional inputs. The other parameters for the model were borrowed from the MATLAB deep learning toolbox examples. The comparison of the performance between the DNN and LSTM network is presented in the result section.

In order to avoid over fitting of the LSTM model, the concept of regularization using drop out layer has been used [71]. In this technique, the inputs of the different nodes in the layer are randomly set to zero causing a change in the underlying network architecture. This helps to avoid over fitting as the network model changes in the each iteration based on drop out probability given as the input to the *dropout(probability)* function. In this case, the drop out probability has been set to 20%. This causes 20% nodes in the neural layers to switch off.

4. RESULTS

This section deals with the results of estimations made using the NN. The estimations are made for four temperature cases as mentioned earlier. The temperature cases are $-10°$C, 0°C, 10°C, and 25°C. Initially the estimation was performed using only the voltage(V(t)), current (I(t)) and temperature (T(t)). However, the poor performance of the model made it necessary to design a second model that had two additional inputs to that of previous models. As mentioned earlier, the two additional inputs were the average voltage (Vavg(t)) and average current (Iavg(t)) taken from 500 previous samples. And lastly, an additional network with LSTM layer was built and used for estimating the SOC.

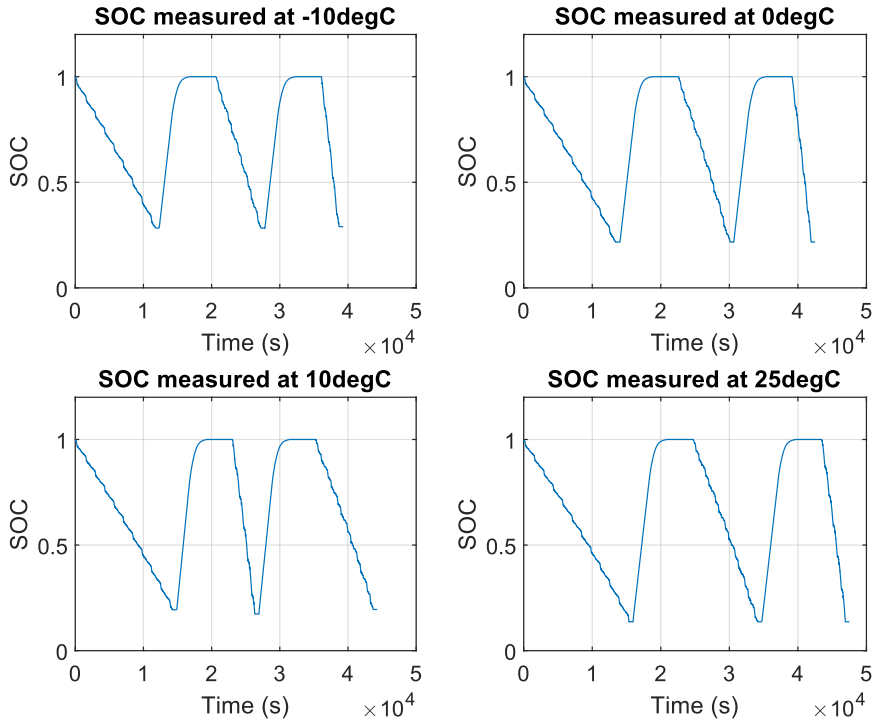

Figure 4.1. *Ground truth for the model*

The SOC for four temperatures collected from the experiment is shown in figure 4.1. These values are the ground truth for the NN model and were used as the test data after the model was trained. As shown in the figure 4.1, the discharge curve are different because of the different drive cycles that was used to discharge the battery during the experiment.

4.1 Estimation using Deep Neural Network

The NN models estimate the SOC based on the training data. Figure 4.2 shows the estimations made at four temperature cases. From the initial observation , the model performs well in following the SOC curve 'pattern'. The model was initially trained with the dataset which had a mix of all temperature data and then tested on data from different temperature cases. Thus, it can be understood that the model was able to generalize and didn't memorize the training data. The four subplots represent the four different temperature cases.

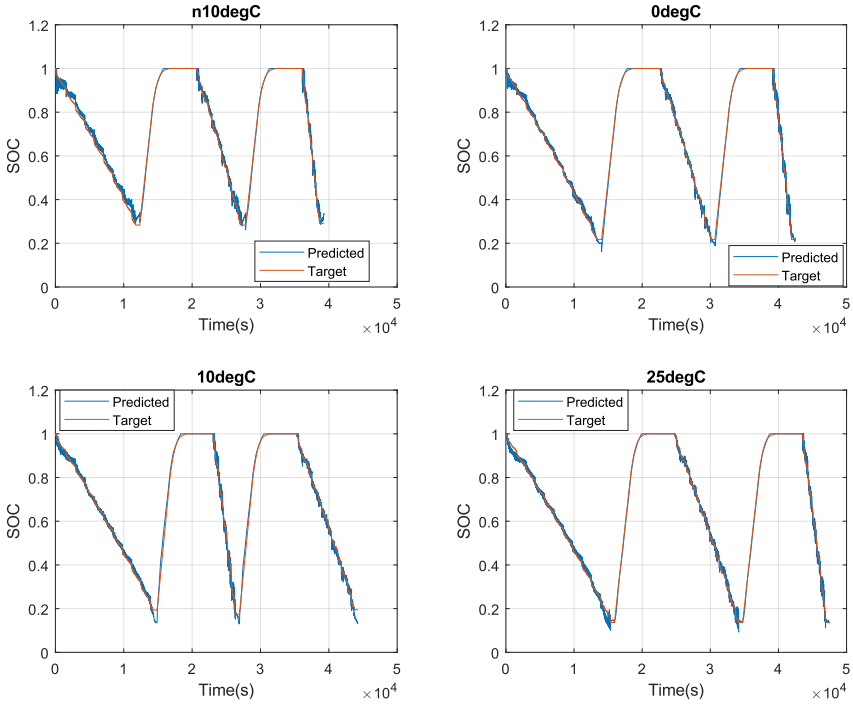

Figure 4.2. *Plot of estimated SOC for four different temperatures using DNN*

Error	-10°C	0°C	10°C	25 °C	Average
MAE(%)	1.20	0.88	1.06	0.84	0.99
RMSE(%)	1.65	1.26	1.63	1.26	1.45
Max (%)	24.06	24.383	30.049	40.44	29.73

Table 4.1. *Error metrics using DNN at different temperatures (Epochs: 1000)*

The error metrics in figure show the estimation error for the four temperature cases. The error metrics considered here are the Mean Absolute Error(MAE), Root Mean Square Error (RMSE) and Maximum percentage error (Max), all expressed in percentage.

The maximum error is observed at 25°C. Figure 4.4 shows the SOC estimation at 25°C, along with the percentage error. It appears that the model's estimation diverges signifi-

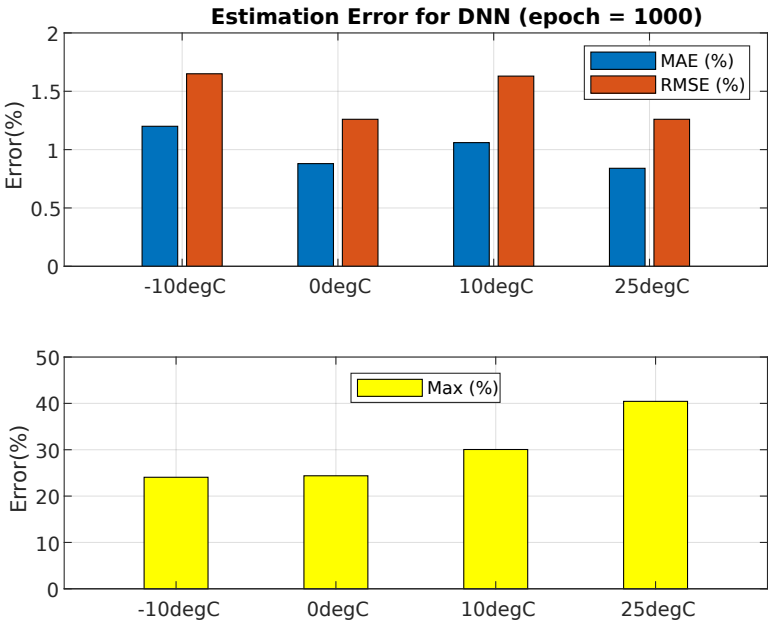

Figure 4.3. *Estimation error for DNN after 1000 epochs*

cantly during the last moments of the discharge curve when the SOC is approaching zero. Thus, the maximum error case is observed at these points. One reason for this anomaly is the averaging of the previous time steps data. When the mode of battery operation changes from charging to discharging, the data from previous mode isn't relevant as the direction of current flow and hence the SOC both change. One option could be to develop different NN model for charging mode and discharging mode. Further discussion on these issue is presented in next section. The graphs for all other temperature cases is shown in Appendix A.

To mitigate these errors, one alternative approach would be to train the model for more epochs with more neurons in the hidden layer. This gives the model more iterations to adjust the weights and learn the training data well enough such that the estimation error is minimized. Vidal et al.[65] suggested using 5100 epochs and 100 neurons to get better results. Adding more neurons per hidden layer and increasing the epochs however increases the training time and the risk of over-fitting. Hence, the design choice has to be made between number of NN parameters and accuracy. Based on the figure 3.7, 97 is the highest value of neurons that can be chosen to get minimal error. Hence, model was run for 5100 epochs and number of neurons per hidden layer was changed to 97.

It was observed that the error decreased as the number of neurons per hidden layer and number of epochs was increased. However, the decrease in error is not significantly high. Table 4.2 shows the error metrics when the model was trained for 5100 epochs with 97 neurons per hidden layer.

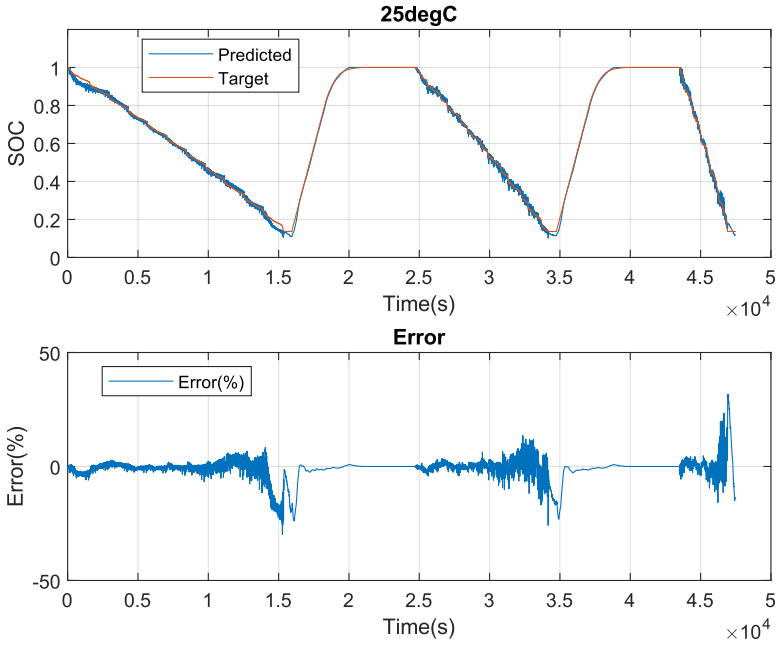

Figure 4.4. *SOC Estimation with percentage error after 1000 epochs at 25° C*

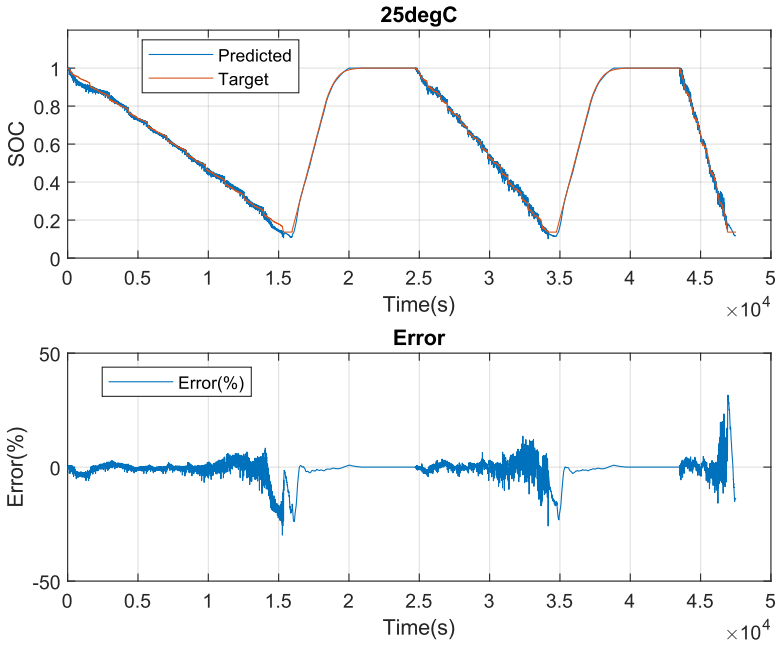

Figure 4.5. *SOC Estimation with percentage error after 5100 epochs at 25 ° C*

4.2 Estimation using LSTM Network

Another model consisting of LSTM layer was prepared for estimation of SOC. The main motivation behind this was to observe if this model performs any better compared to Deep Neural Network. This model works on the feedback loop and is able to keep track of previous outputs of the hidden layer and adjust it's current time-step output depending on

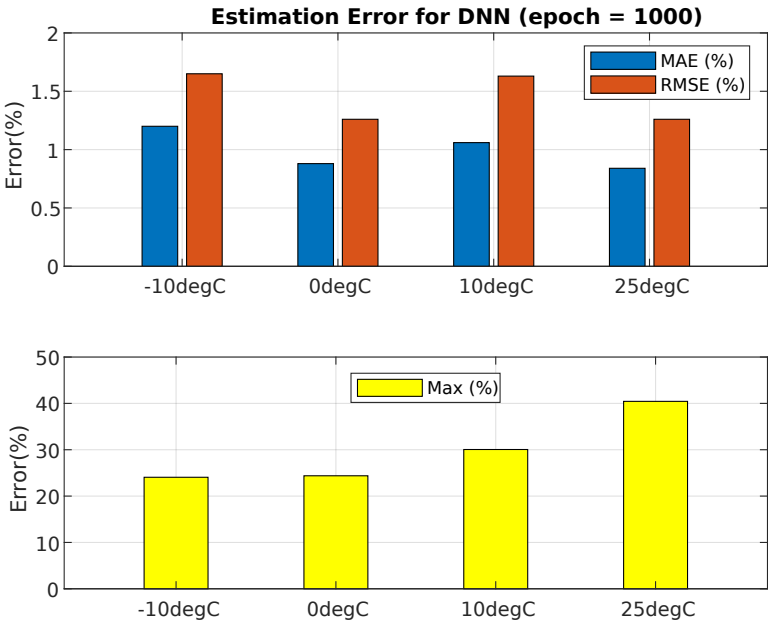

Figure 4.6. *Estimation error for DNN after 5100 epochs*

Error	-10°C	0°C	10°C	25 °C	Average
MAE(%)	1.13	0.88	1.06	0.81	0.97
RMSE(%)	1.56	1.25	1.60	1.23	1.41
Max (%)	19.82	25.55	26.624	38.36	27.59

Table 4.2. *Error metrics using DNN at different temperatures (Epochs: 5100)*

these previous input. This network is a type of Recurrent Neural Network (RNN) and has been used to estimate the SOC of Lithium ion cells [8]. Figure 4.7 shows the estimation of SOC performed by a LSTM network.

The model has worked well in picking up the discharge profile pattern. As can be seen from the figure 4.8, the maximum estimation error is seen in the 10°C. The estimation made for 10°Cis shown in the figure 4.9 .

The maximum error in this case is observed during the discharging time just after the SOC was minimum. The estimation curve closely resembles the discharge profile with MAE of 1.08%. Figure 4.9 shows the estimation error for 10°Cdata. It can be observed that the maximum error in this case is also around the low SOC points where the discharge cycle has just ended and the charging cycle begins. Apart from that, the model follows the discharge profile with average MAE of 1.08%. It should be however noticed that the number of epochs that the LSTM network has been trained for is just 200. The reason for using such small epochs is because the LSTM network takes much longer time to train compared to DNN for same number of neurons. The memory effect and the use of forget gate causes the network to have multiple operations to be carried during one iteration.

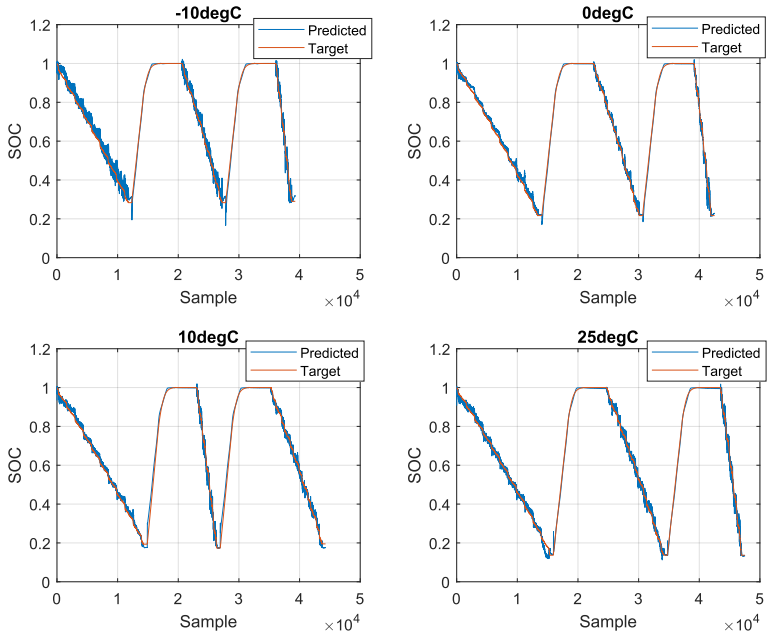

Figure 4.7. *Plot of estimated SOC for four different temperatures using LSTM network*

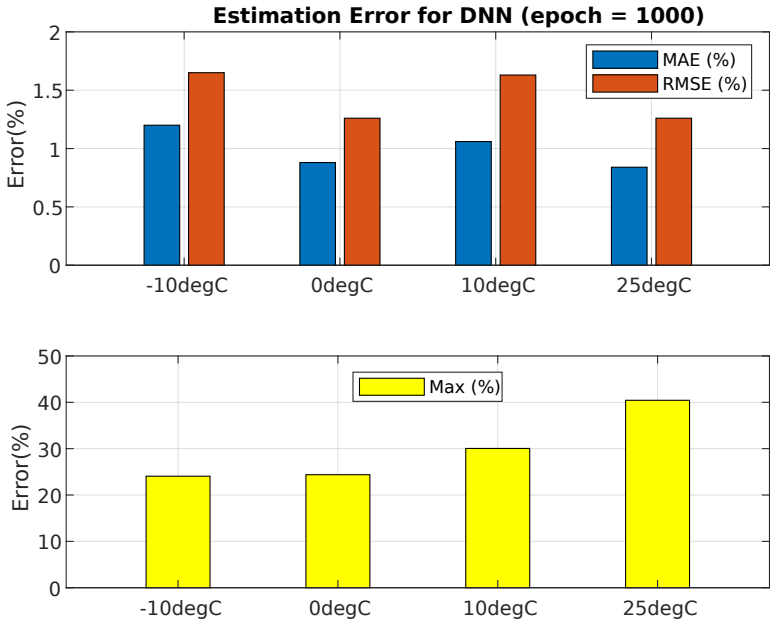

Figure 4.8. *Estimation Error for LSTM network*

Hence, this takes longer time. With powerful machines and higher number of iterations, LSTM networks can prove much more effective than DNN network. The graphs for all other temperature cases is shown in Appendix A.

One of the reasons for high error when the battery mode changes from charge to discharge and vice versa is the discontinuous measurement.There are four modes: charging, idle after charging, discharging and idle after discharging. This cause the model to

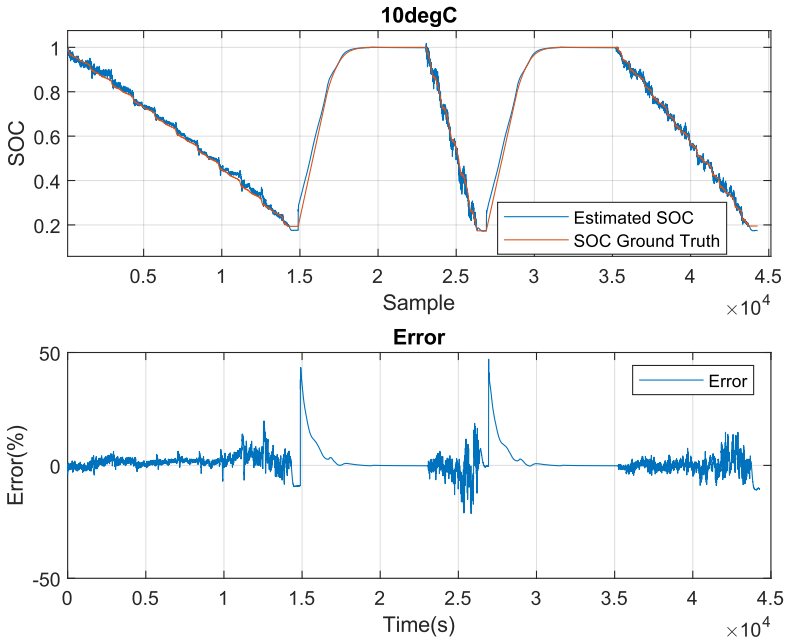

Figure 4.9. *Plot of estimated SOC for 10° C*

Error	-10°C	0°C	10°C	25 °C	Average
MAE(%)	1.27	1.02	1.23	0.79	1.08
RMSE(%)	1.87	1.44	1.89	1.19	1.6
Max (%)	31.28	21.62	47.29	40.72	35.22

Table 4.3. *Error metrics using LSTM at different temperatures (Epochs: 200)*

partially look up for data representing previous mode (for instance discharging) when the current mode is different (for instance, charging). Also, the average voltage and current taken from previous time steps affects the estimation during this boundary of charge and discharge. And in case of LSTM networks, long feedback from previous mode messes up the estimation for the present mode which is why the error is maximum in the moments when the mode of battery is changing. Further explanation about this is done in the next section.

5. DISCUSSION

This thesis dealt with the use of Artificial neural network models to estimate the SOC of the Lithium ion battery. Figure 4.2 and figure 4.7 present the estimation of SOC made by NN. To measure the performance index of the models, three error metrics namely, mean absolute error, root mean square error and maximum error were observed for each model. The DNN was trained for 1000 epochs initially. The DNN was also trained for higher number of epochs to observe whether there would be significant performance differences or not. Next, a Long Short Term Memory Network (LSTM) was utilized to contrast the performance with DNN. This is because the two networks differ in architecture and design. Observing performance of both the models helps in understanding ANN design for BMS in great detail. All these models were compared with each other to see, which model performed the best. The summary of the models is presented in the table 5.1. The main parameters in consideration for comparison are the RMSE, MAE , max error and training time.

Items	DNN (1000 epochs)	DNN (5000 epochs)	LSTM (200 epochs)
MAE_{avg} (%)	0.99	0.97	1.08
$RMSE_{avg}$ (%)	1.45	1.41	1.6
Max_{avg} (%)	29.73	27.59	35.22
Training Time	5 minutes	43 minutes	13 minutes

Table 5.1. *Summary of the model performances*

It was observed that DNN trained for 5000 epochs gave the least error and took the longest time to train. However, LSTM layer was able to provide comparable performance with significantly lower number of neurons and training time. This also provides convincing evidence that LSTM layer can be one of the best options as a NN model for SOC estimation.

Based on the experiments and the obtained results, some interesting observations have been made. The NN model, unsurprisingly, seems to work better with more number

of epochs and higher number of neurons per hidden layer. However, the trade-off is the training time. With more neurons in hidden layer and more epochs, the machine takes longer time to train the model. One solution could be to use different optimization algorithms to set up the hyper parameters that could speed up the training process. Also using dedicated processing units like GPU (Graphical Processing Unit) and NPU (Neural Processing Unit) can decrease the training time.

Comparing the DNN model with LSTM model, it has been observed that with multiple epochs and higher number of neurons, DNN seems to outperform LSTM network. However, it should be considered that LSTM network takes a long time to train owing to the memory effect and hence, have been trained for relatively less number of epochs compared to DNN (due to hardware constraints). But with higher epochs, LSTM network should be able to outperform Feedforward DNN in estimation of SOC.

Another approach could be to update some layers during the training process and keep other layers unchanged (freeze) and update them later albeit less frequently. This can reduce the computational burden as the number of layers to work per epoch are reduced. Brock et al. [5] present an interesting finding regarding the freezing of hidden layers in DNN. They found that they could reduce the training speed by 20% without any loss of accuracy for Residual Neural Networks (ResNets). Similar, approach might find application with this model too.

It has also been noticed that the error metric change slightly at each training process. It is because the NN are initialized with random weights at the beginning of training process and hence result in a slightly different error metrics after 1000 epochs. However, this change is quite low and doesn't affect the over all analysis of the models.

The model seems to suffer from poor estimations as the SOC approaches zero during the discharge profile. This was observed in all the temperature estimations cases. This is due to change in the two discrete events i.e. charging and discharging. As, discharging is completely different event, it doesn't take into consideration the charging case. And hence, the large errors in the moment when this mode changes. Using different neural network to model charging and discharging cases is one simple way to tackle this issue.

One of the reasons could be the inefficient design of the model itself. Tweaking the hyper parameters is one way to optimize the model and minimize this anomaly. Training with larger dataset than the one available now or using pre-trained models developed by big institutions might also reduce this anomaly. A larger dataset offers the neural network additional examples and diverse data variations. This aids in fostering a deeper understanding of generalized patterns, rather than merely memorizing specific instances from a smaller dataset. Big institutions have the resources to train models on bigger data sets with state of the art computing hardware and software that can cost a large sum of money and hence is only affordable for big institutions. Another reason could be

that the voltage and current data obtained during these moments could be inconclusive enough to estimate the SOC accurately. Hence, extra parameters might be necessary to estimate SOC more accurately. Also, the effect of different C-rates affects the batteries charge holding capacity [41]. This also causes them to behave erratically when they are discharged deeply. While these reasons are not the final verdict on the anomaly observed, these can be the starting point for future research endeavors in this field.

The input vector consists of only the voltage, current and temperature in the case of this thesis. While these parameters are paramount for determining the SOC of the cell, it has been observed that these are not the sole factors to consider for accurately assessing the SOC. There are other parameters for instance, the history of the cell, manufacturing material and style, age of the cell, cycle count, depth of discharge etc. that affect the SOC of the cell. Hence, expanding the input vector to encompass these relevant parameters can help in improving the efficiency of the model and make it robust for SOC prediction in the future.

Among other parameters, aging of the cells drastically affect the SOC and is one of the most fundamental topic of research in battery community. Most of the research on application of AI for SOC estimation ignore the aging of the battery due to the difficulty in quantifying it effect. Aging is slow for unused batteries. But batteries that have been discharged deeply and batteries that are used in very low temperatures for long period of time undergo aging faster compared to batteries that are stored. Also, the depth of discharge history and the history of the cycle count plays an important role in determining the SOC of the battery. The model in this thesis has not considered these parameters because of their pragmatic nature and difficulty in recreating those scenarios in the laboratory for testing. Hence, to model a complete battery, data from a complete life-cyle of a battery is necessary along with other parameters that are mentioned above. This will provide an inclusive database for the NN model to work increasing their generalizing capability. Cheng et al. [11] have researched onto this issue to co-relate aging with SOC estimation. Improving upon this model can be a new avenue of research for the future work.

Besides, various random environmental, human or technical variable that is uncounted for when developing the models can pose an issue if applied to real life cases. For example EV batteries working in difficult terrains might not follow the drive cycles used to analyze the test cell and hence these models might prove ineffective for such cases. The model in this thesis has been prepared on the basis of a cylindrical cell of 18mm diameter and 65mm height (also referred to as 18650 cells in battery industry). There are multiple other cell types of Lithium-ion chemistry like prismatic, pouch, button etc. which have not be considered. Generalizing this to even larger scale, SOC predictions for all cell chemistries can also be an area of research for the future.

The whole premise of this thesis was build on the idea that with the increase connectivity and automation, large database of battery information can be built and then be used to train the NN models to accurately predict the SOC. Implementing strategies that can help in accumulating these information can be a step forward in realizing this battery database. Cloud storage of battery information from EV modules, Battery Energy Storage Systems (BESS), battery manufacturers, EV manufacturers, EV maintenance companies, EV users, etc. can help in gathering these information for this proposed battery database. This can help in generating a mega-battery-database which can then be used to train the NN models in the future. Pre-trained model can also be provided by institutions working closely with Lithium-ion batteries.

Lastly, even though the SOC estimation with ANN was achieved, there are many underlying assumptions that has been used to simplify the analysis. Some assumptions like modelling only for some temperature cases are due to resource constraints like lack of data availability and need of powerful processing units to model large dataset. Other assumptions like the use of drive cycle to model EV was due to practical reasons. As it is really difficult to model every user behaviour and thus the discharge characteristics of the battery in EV modules. Hence, freely available standard models provided by EPA has been used to model EV discharge pattern in this case. Another assumption lies in ignoring the historical charge/discharge characteristics. It is a vital piece of information to increase efficiency of NN. Besides, use of drive cycles test data for training and testing the model, using only voltage, current and temperature to make predictions of SOC, not considering the aging of the cell and referring to the cycle count are some of these assumptions. Future research on this topic can focus on these aspects of problems to make SOC estimation much more robust and applicable for daily usage.

This test is a small step to the design of neural network integrated BMS. Currently, SOC estimation techniques in the market is a combination of neural network, Coulomb Counting, Open Circuit Voltage (OCV) and other estimation techniques mentioned in previous sections. Industrial techniques are more or less closely guarded secrets and hence accurately stating what modern EVs use for SOC estimation is hard. Similar is the case with the design of BMS. EV manufacturers are secretive of their technology due to cut-throat competition. But design wise, realizing a BMS integrated with neural network would require several key steps.

1. Data Collection: Gather data from various sources such as battery sensors, voltage and current measurements, temperature sensors, and historical charging and discharging data.
2. Data Preprocessing: Clean and preprocess the collected data to remove noise, outliers, and inconsistencies. This may involve normalization, feature scaling, and

feature extraction to prepare the data for input into the neural network.

3. Neural network Architecture: Design and implement a neural network architecture suitable for BMS applications. This may include selecting the appropriate type of neural network (e.g., feedforward, recurrent, convolutional) and defining the number of layers, neurons, and activation functions.
4. Training: Train the neural network using the preprocessed data. This involves optimizing the network's parameters (e.g., weights and biases) using techniques such as gradient descent and back propagation to minimize the prediction error.
5. Model Evaluation: Evaluate the trained neural network's performance using validation data and metrics such as accuracy, precision, recall, and F1-score. Iterate on the model architecture and training process as needed to improve performance.
6. Integration with BMS: Integrate the trained neural network model into the BMS architecture, allowing it to make real-time predictions and decisions based on incoming sensor data. This may involve deploying the model on embedded hardware or in a cloud-based environment, depending on the application requirements.
7. Testing and Validation: Test the integrated BMS system in simulated and real-world scenarios to ensure its reliability, accuracy, and safety. Validate the system's performance against established benchmarks and industry standards.
8. Deployment and Maintenance: Deploy the BMS system in operational environments, monitoring its performance and conducting regular maintenance to ensure continued functionality and reliability. Update the neural network model as needed to adapt to changing operating conditions or improve performance over time.

It is possible to develop an adaptive neural network model tailored to specific user behavior, EV battery characteristics, and battery health. This would involve training the neural network using data collected from the specific EV, the user's driving patterns, and the battery's performance over time.

The adaptive nature of the neural network could be achieved through techniques such as online learning, where the model continuously updates and refines its parameters based on new data inputs. Additionally, the neural network could incorporate features such as reinforcement learning to adapt its behavior based on user feedback and environmental conditions.

By customizing the neural network model to the individual user's behavior and the characteristics of their EV battery, it could provide more accurate predictions and recommendations for optimizing battery usage, extending battery life, and enhancing overall vehicle performance.

6. CONCLUSION

In this thesis, Artificial Neural Networks were considered as an alternative tools for SOC estimation of Li-ion batteries. Conventional methods for estimating State of Charge (SOC) frequently encounter difficulties due to the dynamic and intricate nature of battery behavior across various conditions, resulting in inaccuracies and diminished performance. Conversely, machine learning methods, harnessing data-driven insights, demonstrate superior adaptability to such challenges. Namely, two ANNs, Deep Neural Network (DNN) and Long Short Term Memory Network (LSTM) were used to estimate the SOC of an LG 18650 HG2 cell for four temperature cases. These were -10°C, 0 °C, 10°Cand 25°C. The DNN was trained initially with only three inputs viz. voltage, current and temperature. However, the estimation performance with these inputs was not satisfactory. Referring to Vidal et al. [65], along with these three parameters, two extra inputs namely, average voltage and average current were also used to make estimations. These average value were calculated from 500 previous time-step values. This significantly improved the performance of the DNN. The average MAE of the DNN was found to be around 0.99%. The performance of the DNN was also observed by increasing the number of neurons and training epochs to see if there would be any significant improvement in the performance. It was observed that while the performance was improved, as MAE in this case was 0.97%, it was not significantly high.

Next, a different type of ANN called LSTM was used to test the SOC estimation capability. LSTM are a type of recurrent neural networks where the information flows in both the forward and backward directions. LSTM is a special class of RNN where the network can keep track of previous inputs using its memory cells. It has a forget gate along with the input and output gate, that makes it useful for predicting time series data. However, because of this memory gate, LSTM network take longer time to train compared to DNN for the same number of neurons. Because of this, LSTM network have been trained for far less number of epochs than DNN. Despite this, LSTM network performed competitively. For just 200 epochs, the average MAE for the estimation was 1.08%. Further research can be done to improve the performance of the LSTM network and make it more robust.

The objective of this thesis was to see the effectiveness of the neural network models in estimating the SOC of the Lithium-ion battery. Two instance of ANN, namely, DNN and LSTM were used for this task for their simplicity and ease of use. It was observed that the

larger the dataset the better was the performance of the model. ANN was indeed able to handle non-linear relationship between input parameters of battery (Voltage, Current, and Temperature) and SOC with good accuracy. The risk of over-fitting was avoided by splitting the initial data set early in the data preparation stage. And also by using least number of model parameters and not over training the model for large number of epochs such that it doesn't memorize the specific instances form the training data set. One observation that was made during the thesis work was that discrete battery operation modes i.e. charging and discharging behave differently and one single ANN model will struggle during these mode change. However, using different model for charging and discharging cases can be one solution to this problem.

While ANN have proved competent as an alternative modelling techniques for SOC estimation, they require large data base of cell parameters to work accurately. However, with the increase in internet connectivity and automation, a global database of different cell characteristics can be realized soon. And with the increase in computational power and dedicated processing units like NPU (Neural Processing Units), more complicated NN can be realized which shall improve the accuracy of SOC estimation in the future.

REFERENCES

[1] International Energy Agency. *Global EV outlook 2023*. Tech. rep. International Energy Agency, 2023.

[2] Yūichirō Anzai. *Pattern recognition and machine learning*. eng. 1st edition. Boston: Academic Press, 1992. ISBN: 0-08-051363-8.

[3] Charlotte Argue. *How long do electric car batteries last? What 6,300 electric vehicles tell us about EV battery life*. 31-05-2024. URL: https://www.geotab.com/blog/ev-battery-health/.

[4] Christopher M. Bishop. *Pattern recognition and machine learning*. eng. Information science and statistics. New York: Springer, 2006 - 2006. ISBN: 0-387-31073-8.

[5] Andrew Brock et al. "Freezeout: Accelerate training by progressively freezing layers". In: *arXiv preprint arXiv:1706.04983* (2017).

[6] Hicham Chaoui, Chinemerem C Ibe-Ekeocha, and Hamid Gualous. "Aging prediction and state of charge estimation of a LiFePO4 battery using input time-delayed neural networks". In: *Electric Power Systems Research* 146 (2017), pp. 189–197.

[7] Mohammad Charkhgard and Mohammad Haddad Zarif. "Design of adaptive H filter for implementing on state-of-charge estimation based on battery state-of-charge-varying modelling". In: *IET Power Electronics* 8.10 (2015), pp. 1825–1833.

[8] Ephrem Chemali et al. "Long short-term memory networks for accurate state-of-charge estimation of Li-ion batteries". In: *IEEE Transactions on Industrial Electronics* 65.8 (2017), pp. 6730–6739.

[9] Ephrem Chemali et al. "State-of-charge estimation of Li-ion batteries using deep neural networks: A machine learning approach". In: *Journal of Power Sources* 400 (2018), pp. 242–255.

[10] Jian Chen et al. "Neural network-based state of charge observer design for lithium-ion batteries". In: *IEEE Transactions on Control Systems Technology* 26.1 (2017), pp. 313–320.

[11] Ming-Wang Cheng et al. "State-of-charge estimation with aging effect and correction for lithium-ion battery". In: *IET Electrical Systems in Transportation* 5.2 (2015), pp. 70–76.

[12] Devika Chhachhiya, Amita Sharma, and Manish Gupta. "Designing optimal architecture of neural network with particle swarm optimization techniques specifically for educational dataset". In: *2017 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence*. IEEE. 2017, pp. 52–57.

[13] Martin Coleman et al. “State-of-charge determination from EMF voltage estimation: Using impedance, terminal voltage, and current for lead-acid and lithium-ion batteries”. In: *IEEE Transactions on industrial electronics* 54.5 (2007), pp. 2550–2557.

[14] Andy Colthorpe. *Moss Landing: World’s biggest battery storage project is now 3GWh capacity*. August 2, 2023. URL: https://www.energy-storage.news/moss-landing-worlds-biggest-battery-storage-project-is-now-3gwh-capacity/.

[15] European Commission. *Li-ion batteries for mobility and stationary storage applications: Scenarios for costs and market growth,* tech. rep. Joint Reseach Center, Petten, 2018.

[16] “Critical review of the methods for monitoring of lithium-ion batteries in electric and hybrid vehicles”. eng. In: *Journal of power sources* 258 (2014), pp. 321–339. ISSN: 0378-7753.

[17] Zhenhua Cui et al. “A comprehensive review on the state of charge estimation for lithium-ion battery based on neural network”. In: *International Journal of Energy Research* 46.5 (2022), pp. 5423–5440.

[18] Bin Ding, Huimin Qian, and Jun Zhou. “Activation functions and their characteristics in deep neural networks”. In: *2018 Chinese control and decision conference (CCDC)*. IEEE. 2018, pp. 1836–1841.

[19] KC Divya and Jacob Østergaard. “Battery energy storage technology for power systems—An overview”. In: *Electric power systems research* 79.4 (2009), pp. 511–520.

[20] Fathima Fasmin and Ramanathan Srinivasan. “Nonlinear electrochemical impedance spectroscopy”. In: *Journal of The Electrochemical Society* 164.7 (2017), H443.

[21] Mingyu Gao, Yuanyuan Liu, and Zhiwei He. “Battery state of charge online estimation based on particle filter”. In: *2011 4th International Congress on Image and Signal processing*. Vol. 4. IEEE. 2011, pp. 2233–2236.

[22] Aurélien Géron. *Neural networks and deep learning*. eng. 1st edition. Beijing: O’Reilly Media, Inc., 2018. ISBN: 1-4920-3735-4.

[23] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[24] Ivan Gridin. *Automated deep learning using neural network intelligence : develop and design PyTorch and TensorFlow models using Python*. eng. New York, New York: Apress L. P., 2022. ISBN: 1-4842-8149-7.

[25] Shanshan Guo and Liang Ma. “A comparative study of different deep learning algorithms for lithium-ion batteries on state-of-charge estimation”. In: *Energy* 263 (2023), p. 125872.

[26] Mohammad A Hannan et al. “A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: Challenges and

recommendations". In: *Renewable and Sustainable Energy Reviews* 78 (2017), pp. 834–854.

[27] Wei He et al. "State of charge estimation for electric vehicle batteries using unscented kalman filtering". In: *Microelectronics Reliability* 53.6 (2013), pp. 840–847.

[28] Zhiwei He et al. "A joint model and SOC estimation method for lithium battery based on the sigma point KF". In: *2012 IEEE Transportation Electrification Conference and Expo (ITEC)*. IEEE. 2012, pp. 1–5.

[29] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[30] Dickshon NT How et al. "State-of-charge estimation of li-ion battery in electric vehicles: A deep neural network approach". In: *IEEE Transactions on Industry Applications* 56.5 (2020), pp. 5565–5574.

[31] University of Washington Clean Energy Institute. *What is a lithium-ion battery and how does it work?* Accessed: 14/12/2023.

[32] Xiaosong Hu et al. "Online estimation of an electric vehicle lithium-ion battery using recursive least squares with forgetting". In: *Proceedings of the 2011 American Control Conference*. IEEE. 2011, pp. 935–940.

[33] Yuhuang Hu et al. "Overcoming the vanishing gradient problem in plain recurrent networks". In: *arXiv preprint arXiv:1801.06105* (2018).

[34] Zhelin Huang et al. "Convolutional gated recurrent unit–recurrent neural network for state-of-charge estimation of lithium-ion batteries". In: *Ieee Access* 7 (2019), pp. 93139–93149.

[35] John FC Khaw, BS Lim, and Lennie EN Lim. "Optimal design of neural networks using the Taguchi method". In: *Neurocomputing* 7.3 (1995), pp. 225–245.

[36] Laiqiang Kong et al. "Fast State of Charge Estimation for Lithium-ion Battery Based on Electrochemical Impedance Spectroscopy Frequency Feature Extraction". In: *IEEE Transactions on Industry Applications* (2023).

[37] Emmanouil D Kostopoulos, George C Spyropoulos, and John K Kaldellis. "Real-world study for the optimal charging of electric vehicles". In: *Energy Reports* 6 (2020), pp. 418–426.

[38] Ao Li et al. "A review on lithium-ion battery separators towards enhanced safety performances and modelling approaches". In: *Molecules* 26.2 (2021), p. 478.

[39] Lin Li, Fadwa Dababneh, and Jing Zhao. "Cost-effective supply chain for electric vehicle battery remanufacturing". In: *Applied energy* 226 (2018), pp. 277–286.

[40] Mingheng Li. "Li-ion dynamics and state of charge estimation". In: *Renewable Energy* 100 (2017), pp. 44–52.

[41] Guangming Liu et al. "Analysis of the heat generation of lithium-ion battery during charging and discharging considering different influencing factors". In: *Journal of Thermal Analysis and Calorimetry* 116 (2014), pp. 1001–1010.

[42] Languang Lu et al. “A review on the key issues for lithium-ion battery management in electric vehicles”. In: *Journal of power sources* 226 (2013), pp. 272–288.

[43] Adnan M. “The Future of Energy Storage: Advancements and Roadmaps for Lithium-Ion Batteries”. In: *International Jounral of Molecular Science* 8.7457 (2023), p. 24.

[44] Mehrdad Mastali et al. “Battery state of the charge estimation using Kalman filtering”. In: *Journal of Power Sources* 239 (2013), pp. 294–307.

[45] Jinhao Meng, Guangzhao Luo, and Fei Gao. “Lithium polymer battery state-of-charge estimation based on adaptive unscented Kalman filter and support vector machine”. In: *IEEE Transactions on Power Electronics* 31.3 (2015), pp. 2226–2238.

[46] Phillip Kollmeyer, Carlos Vidal and Mina Naguib. *LG 18650HG2 Li-ion Battery Data and Example Deep Neural Network xEV SOC Estimator Script*. https://data.mendeley.com/datasets/cp3473x7xv/3. Accessed: 2023-12-21. 2020.

[47] Gregory L Plett. “Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 1. Background”. In: *Journal of Power sources* 134.2 (2004), pp. 252–261.

[48] Gregory L Plett. “Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 3. State and parameter estimation”. In: *Journal of Power sources* 134.2 (2004), pp. 277–292.

[49] Basheer Qolomany et al. “Parameters optimization of deep learning models using particle swarm optimization”. In: *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE. 2017, pp. 1285–1290.

[50] Xiaoqing Ren et al. “A method for state-of-charge estimation of lithium-ion batteries based on PSO-LSTM”. In: *Energy* 234 (2021), p. 121236.

[51] Hannah Ritchie. “Cars, planes, trains: where do CO2 emissions from transport come from?” In: *Our World in Data* (2020). https://ourworldindata.org/co2-emissions-from-transport.

[52] Usha Ruby and Vamsidhar Yendapalli. “Binary cross entropy with deep learning technique for image classification”. In: *Int. J. Adv. Trends Comput. Sci. Eng* 9.10 (2020).

[53] Saeed Sepasi, Leon R Roose, and Marc M Matsuura. “Extended Kalman filter with a fuzzy method for accurate battery pack state of charge estimation”. In: *Energies* 8.6 (2015), pp. 5217–5233.

[54] Sagar Sharma, Simone Sharma, and Anidhya Athaiya. “Activation functions in neural networks”. In: *Towards Data Sci* 6.12 (2017), pp. 310–316.

[55] Iryna Snihir et al. “Battery open-circuit voltage estimation by a method of statistical analysis”. In: *Journal of Power Sources* 159.2 (2006), pp. 1484–1487.

[56] American Chemical Society. *Lithium Ion Battery Recycling*. Nov. 23, 2023. URL: https://www.cas.org/resources/cas-insights/sustainability/lithium-ion-battery-recycling.

[57] Xiaojun Tan. *Battery management system and its applications*. eng. Hoboken, New Jersey: John Wiley Sons, Incorporated, 2023. ISBN: 1-119-15401-4.

[58] Xiaopeng Tang, Yujie Wang, and Zonghai Chen. “A method for state-of-charge estimation of LiFePO4 batteries based on a dual-circuit state observer”. In: *Journal of Power Sources* 296 (2015), pp. 23–29.

[59] Armand M. Tarascon J-M. “Building better batteries”. In: *Nature* 451 (2008), pp. 652–657.

[60] Armand M. Tarascon J-M. “Issues and challenges facing rechargeable lithium batteries”. In: *Nature* 414.6861 (2001), pp. 359–367.

[61] CE Thomas. “Fuel cell and battery electric vehicles compared”. In: *international journal of hydrogen energy* 34.15 (2009), pp. 6005–6020.

[62] United States Environmental Protection Agency. *Dynamometer Drive Schedules*. https://www.epa.gov/vehicle-and-fuel-emissions-testing/dynamometer-drive-schedules. Accessed: 2023-02-19. 2023.

[63] United States Environmental Protection Agency. *Sources of Greenhouse Gas Emissions*. https://www.epa.gov/ghgemissions/sources-greenhouse-gas-emissions. Accessed: 2023-02-09. 2023.

[64] Greg Van Houdt, Carlos Mosquera, and Gonzalo Nápoles. “A review on the long short-term memory model”. In: *Artificial Intelligence Review* 53.8 (2020), pp. 5929–5955.

[65] Carlos Vidal et al. “Robust xev battery state-of-charge estimator design using a feedforward deep neural network”. In: *SAE International Journal of Advances and Current Practices in Mobility* 2.2020-01-1181 (2020), pp. 2872–2880.

[66] Wladislaw Waag, Christian Fleischer, and Dirk Uwe Sauer. “Critical review of the methods for monitoring of lithium-ion batteries in electric and hybrid vehicles”. In: *Journal of Power Sources* 258 (2014), pp. 321–339.

[67] Qiuxia Wang, Peizhou Wu, and Jialing Lian. “SOC estimation algorithm of power lithium battery based on AFSA-BP neural network”. In: *The Journal of Engineering* 2020.13 (2020), pp. 535–539.

[68] Yinjiao Xing et al. “Battery management systems in electric and hybrid vehicles”. In: *Energies* 4.11 (2011), pp. 1840–1857.

[69] Yinjiao Xing et al. “State of charge estimation of lithium-ion batteries using the open-circuit voltage at various ambient temperatures”. In: *Applied energy* 113 (2014), pp. 106–115.

[70] Behnam Zakeri and Sanna Syri. “Electrical energy storage systems: A comparative life cycle cost analysis”. In: *Renewable and sustainable energy reviews* 42 (2015), pp. 569–596.

[71] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. “Recurrent neural network regularization”. In: *arXiv preprint arXiv:1409.2329* (2014).

APPENDIX A: APPENDIX

A.1 Code for DNN

```
%Design the Deep Neural Network


num_input = 5;
num_output = 1;
Epochs = 1000;
mini_batch_size = 1;
LR_drop_period = 200;
initial_LR = 0.01;
LR_drop_factor = 0.1;
Val_frequency = 30;
num_hidden_neurons = 67;
max_error = zeros(1,4);
mean_abs_error = zeros(1,4);
root_mean_square_error = zeros(1,4);
error = zeros(4,3);


%Design the layers of the network
layers = [
sequenceInputLayer(num_input,Normalization="zerocenter")
fullyConnectedLayer(num_hidden_neurons)
tanhLayer
fullyConnectedLayer(num_hidden_neurons)
leakyReluLayer(0.3)
fullyConnectedLayer(num_output)
clippedReluLayer(1)
regressionLayer];


%Set the training options for training the neural network
```

```
train_options = trainingOptions("adam", ...  %algorithm for training=
SGDM (Sotichastic Gradient Descent with Momentum, ADAM, *)
MaxEpochs=Epochs, ...
Plots="training-progress", ...
MiniBatchSize=mini_batch_size, ...
GradientThreshold=1, ...
InitialLearnRate= initial_LR, ...
ValidationData=dsVal, ...
Verbose=false, ...
LearnRateDropPeriod=LR_drop_period, ...
LearnRateDropFactor=LR_drop_factor, ...
LearnRateSchedule="piecewise", ...
ValidationFrequency=Val_frequency, ...
ExecutionEnvironment="gpu", ...
Shuffle = 'every-epoch');


%train the neural network
net = trainNetwork(dsTrain,layers,train_options);


%test the network
YPred = predict(net,tdsPredictorTest,MiniBatchSize=1);
```

A.2 Code for LSTM

```
 clearvars
chunk_size = 500;
num_input = 3;
num_output = 1;
num_hidden_neurons = 256;
Epochs = 400;
mini_batch_size = 64;
LR_drop_period = 30;
initial_LR = 0.01;
LR_drop_factor = 0.1;
grad_thresh = 1;
Val_frequency = 50;
drop_out = 0.2;
```

```
%Design the layers of the network
layers = [ ...
sequenceInputLayer(num_input,Normalization="rescale-symmetric")
lstmLayer(num_hidden_neurons)
dropoutLayer(drop_out)
lstmLayer(num_hidden_neurons/2)
dropoutLayer(drop_out)
fullyConnectedLayer(num_output)
regressionLayer];


%set the training options for training the neural network
train_options = trainingOptions("adam", ...  %algorithm for training=
SGDM (Sotichastic Gradient Descent with Momentum, ADAM, *)
MaxEpochs=Epochs, ...
Plots="training-progress", ...
MiniBatchSize=mini_batch_size, ...
InitialLearnRate= initial_LR, ...
ValidationData=X_val,Y_val, ...
GradientThreshold= grad_thresh, ...
Verbose=false, ...
LearnRateDropPeriod=LR_drop_period, ...
LearnRateDropFactor=LR_drop_factor, ...
LearnRateSchedule="piecewise", ...
ValidationFrequency=Val_frequency, ...
ExecutionEnvironment="gpu", ...
Shuffle = "every-epoch", ...
OutputNetwork = "best-validation-loss");


 %Train the network
net = trainNetwork(X_train,Y_train,layers,train_options);

S = load(test_file);
X_test = S.X(1:  num_input,:);
y_predict = predict(net,X_test);
%Compare estimated SOC with target SOC
Y_test = S.Y;
```

A.3 Estimation Graphs for different Network

For DNN estimations made with 67 neurons and 1000 epochs refer from figure A.1 to A.4.
For DNN estimations made with 97 neurons and 5100 epochs refer from figure A.5 to A.8.
For LSTM network estimations refer from figure A.9 to A.12.

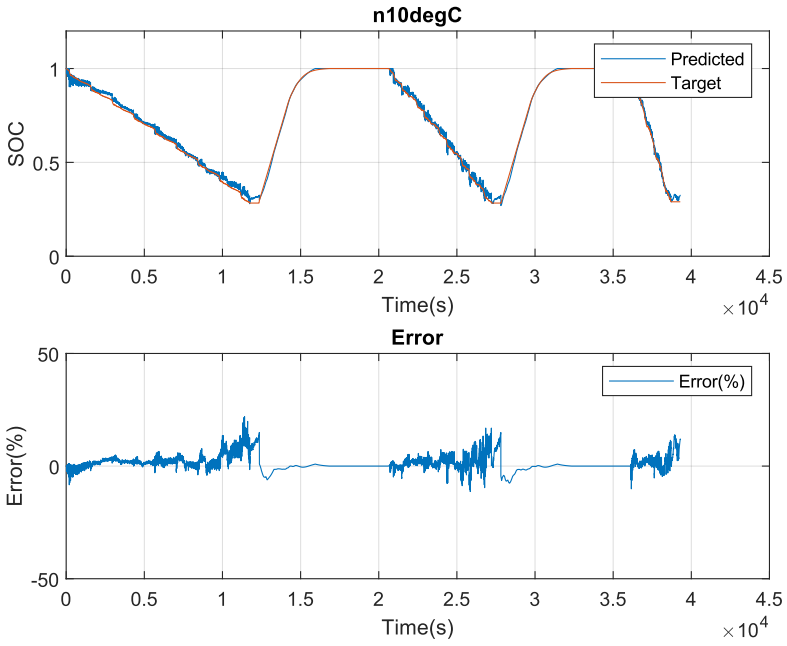

Figure A.1. *SOC Estimation with error at -10 °C*

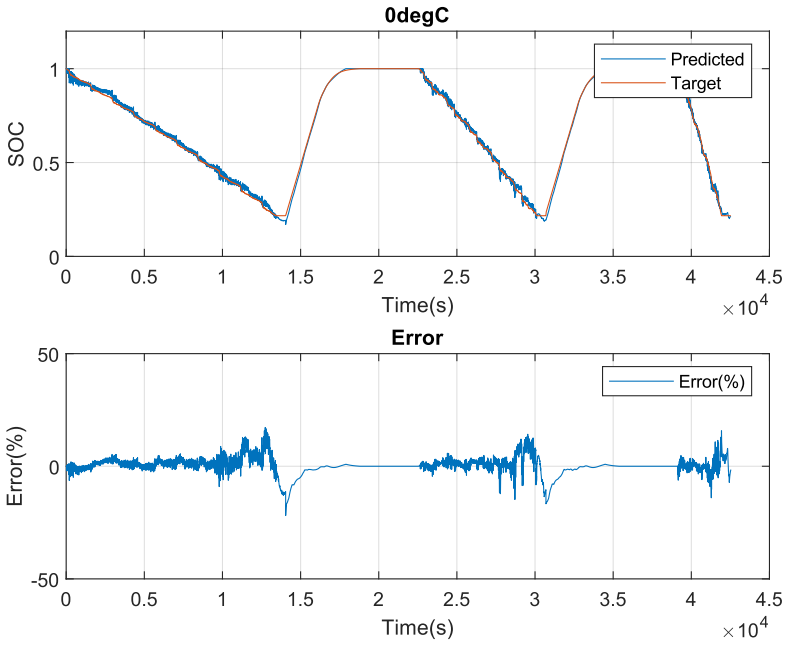

Figure A.2. *SOC Estimation using DNN after 1000 epochs with error at 0 °C*

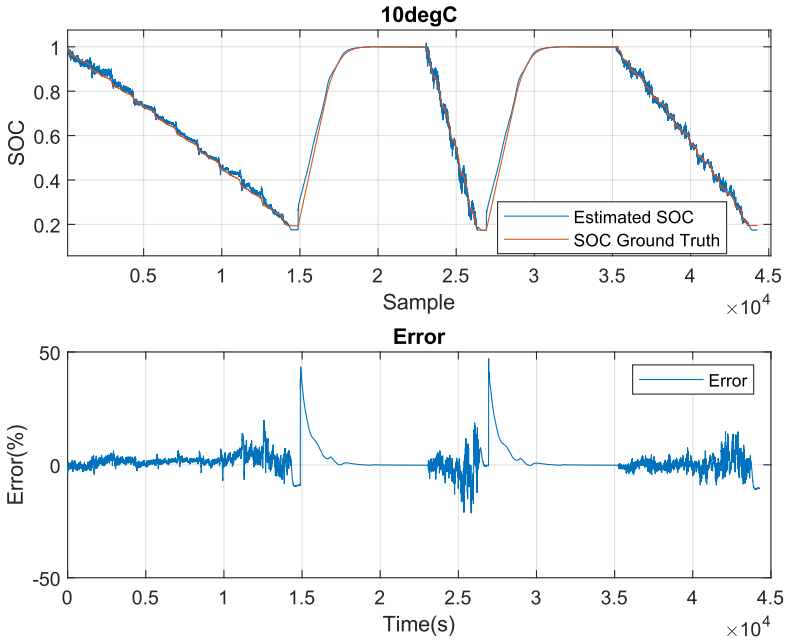

Figure A.3. *SOC Estimation using DNN after 1000 epochs with error at 10 °C*

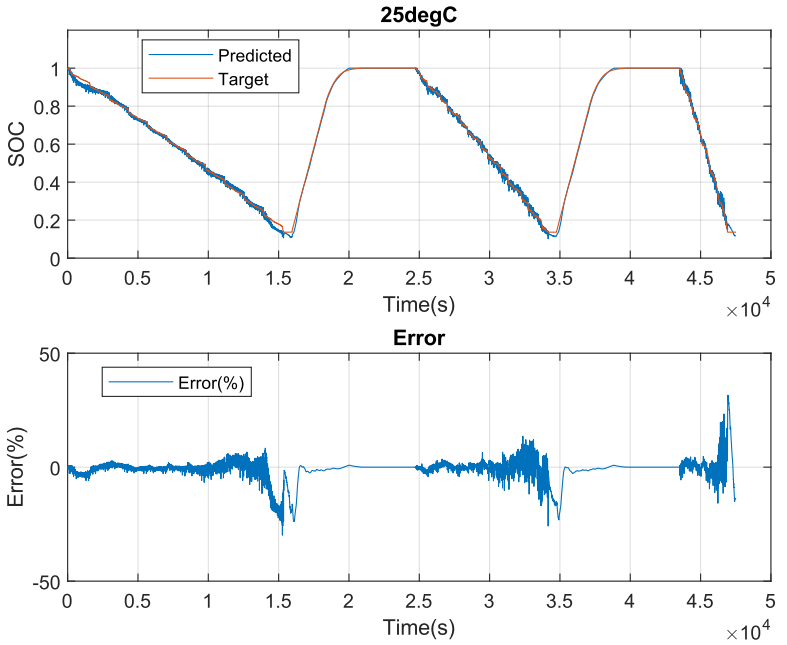

Figure A.4. *SOC Estimation using DNN after 1000 epochs with error at 25 °C*

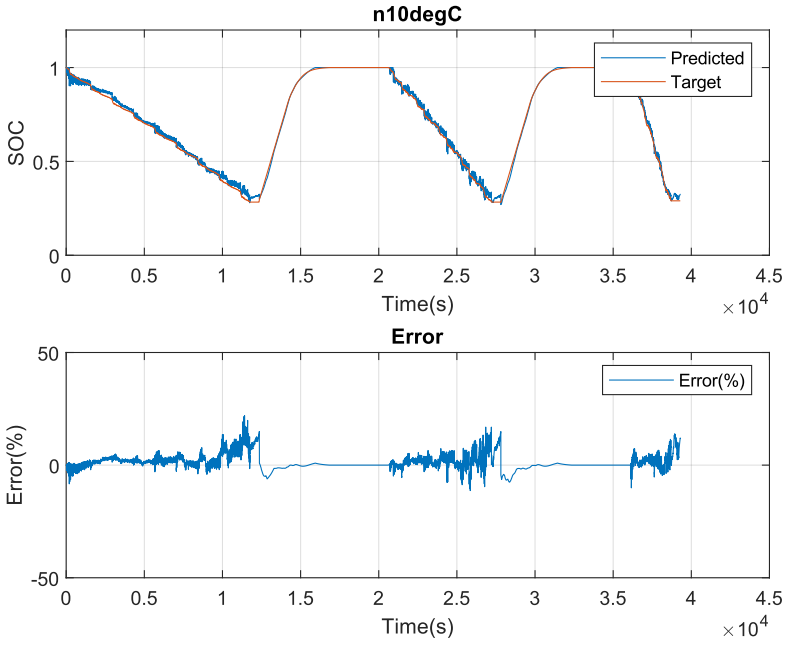

Figure A.5. *SOC Estimation using DNN after 5100 epochs with error at -10 °C*

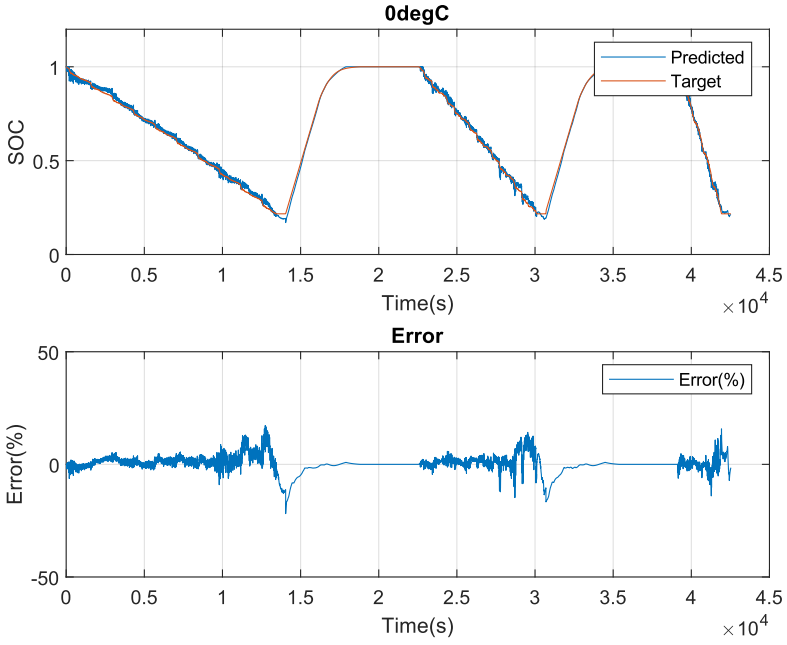

Figure A.6. *SOC Estimation using DNN after 5100 epochs with error at 0 °C*

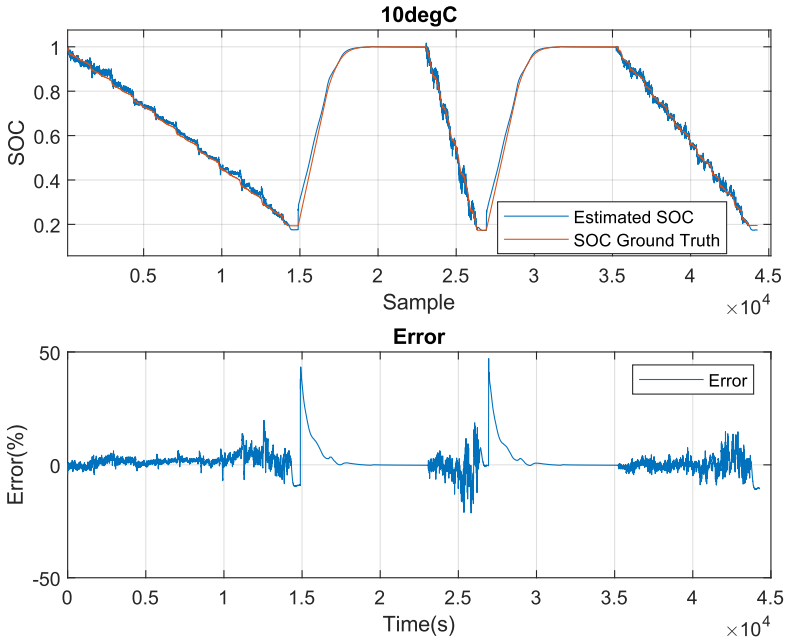

Figure A.7. *SOC Estimation using DNN after 5100 epochs with error at 10 °C*

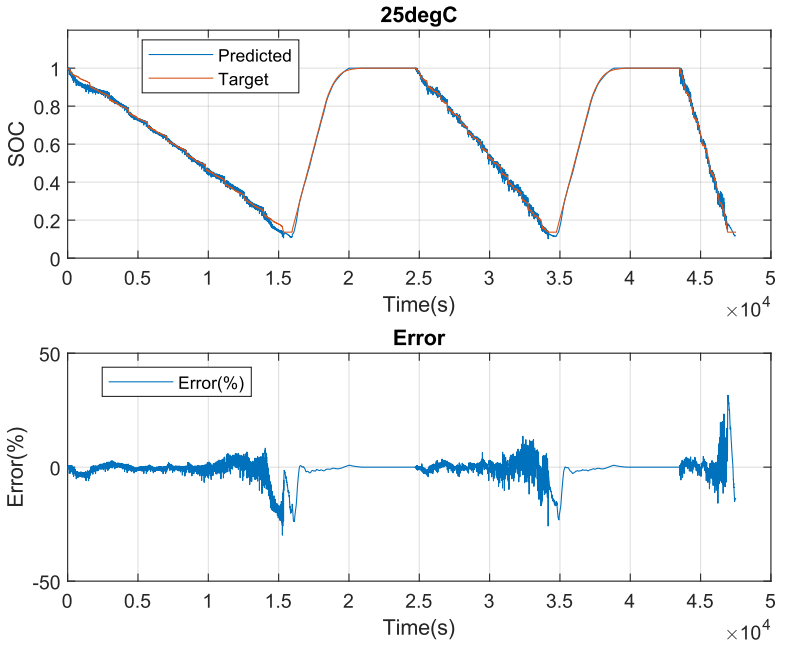

Figure A.8. *SOC Estimation using DNN after 5100 epochs with error at 25 °C*

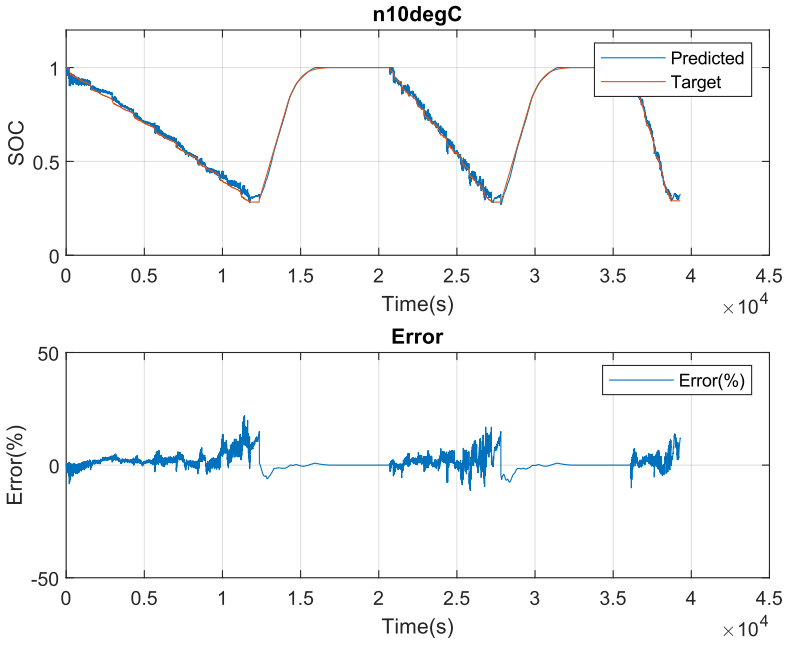

Figure A.9. *SOC Estimation using LSTM after 200 epochs with error at -10 °C*

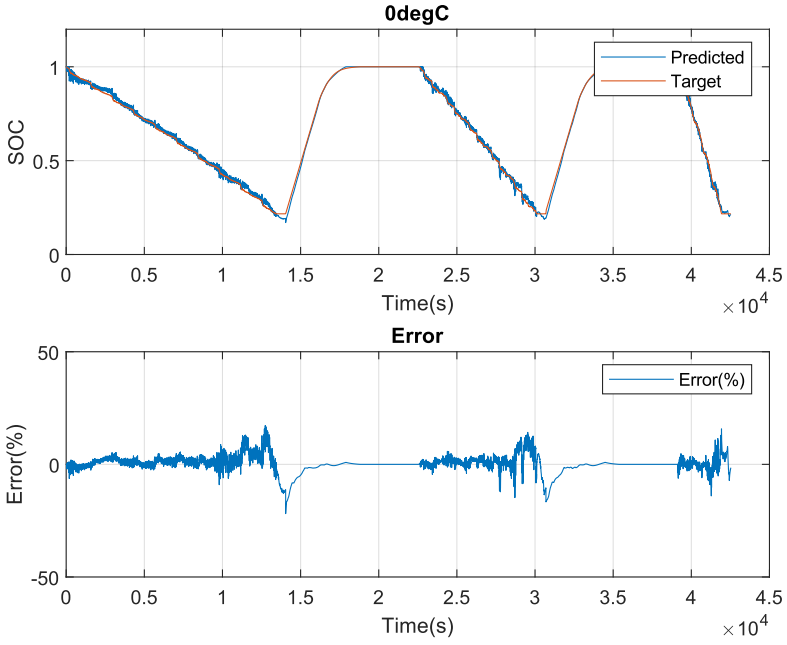

Figure A.10. *SOC Estimation using LSTM after 200 epochs with error at 0 °C*

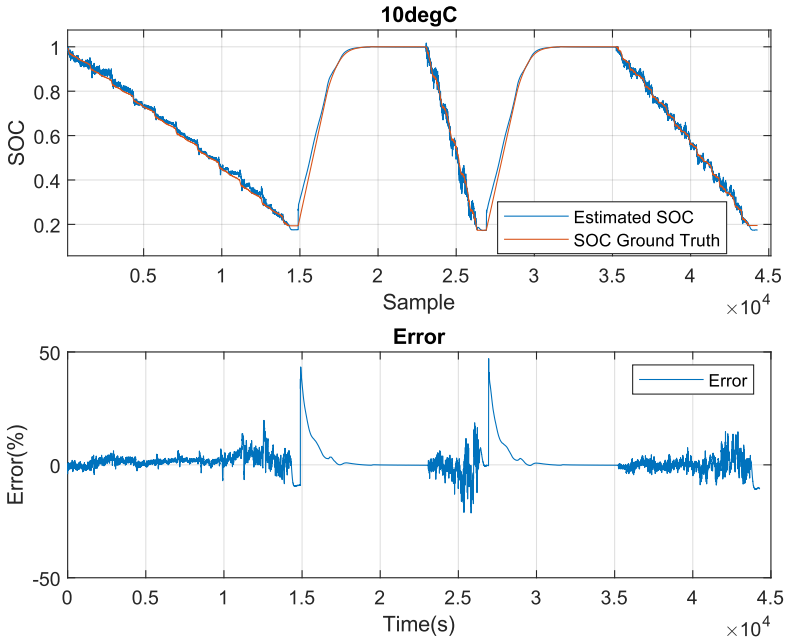

Figure A.11. *SOC Estimation using LSTM after 200 epochs with error at 10 °C*

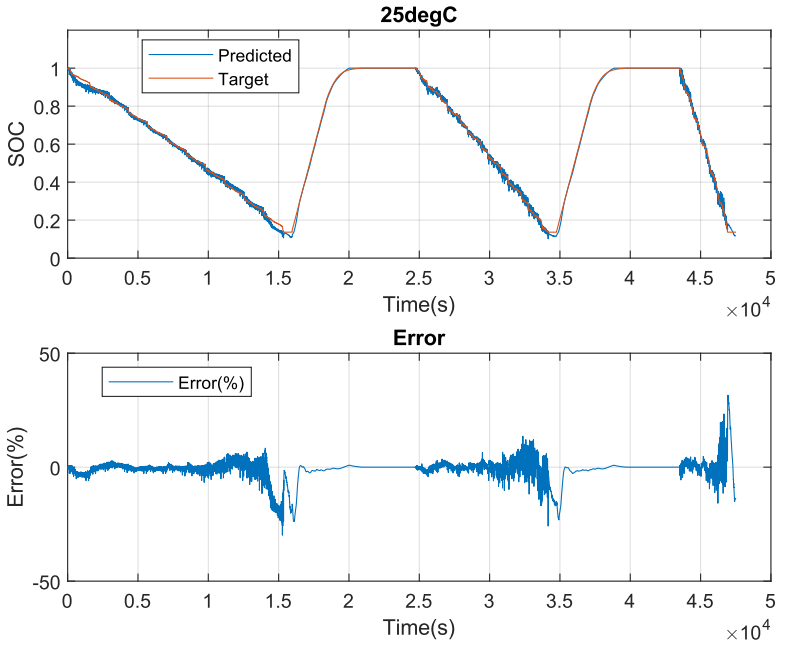

Figure A.12. *SOC Estimation using LSTM after 200 epochs with error at 25 °C*