Proceedings of the 2020 IISE Annual Conference
L. Cromarty, R. Shirwaiker, P. Wang, eds.

Online RNN Model for SOC Prediction in Next Generation Hybrid Car Batteries

Abstract ID: 782506

Steven Hespeler[1], Donovan Fuqua[2]
[1]College of Engineering, [2]College of Business
New Mexico State University, Las Cruces, NM
dfuqua@nmsu.edu

Abstract

This investigation presents a data-driven Long-short Term Memory (LSTM) battery model for predicting State of Charge (SOC) for a lithium-ion battery ($LiFePO_4$) during Electric Vehicle (EV) operation. The LSTM builds and updates a model using multivariate inputs that include physical properties, voltage, current, and temperature during operation. The goal of training is to accurately predict future SOC from multiple training examples using an online learning scheme. Initial results demonstrate excellent prediction with a Root Mean Square Error (RMSE) ranging from $0.372 < RMSE < 0.534$ which outperforms results from literature that utilized other neural network algorithms.

Keywords
Reliability Engineering, Deep Learning, Sustainable System

1. Introduction

With an increase in devices and applications that require batteries, management has become critical to maintaining the safety and reliability of batteries. In the case of Electric Vehicle (EV) operations, a Battery Management System (BMS) controls many parameters. State of Charge (SOC) is widely considered the most influential and important of these parameters and is responsible for safe vehicle operation [1]. SOC is very influential by providing information about the battery's current and remaining life, which is useful for protecting the battery from over-charging/over-discharging. Furthermore, an accurate estimation of the battery state assures reliable and optimal operating conditions for the user. In the case of EV operation, SOC is the equivalent of a fuel gauge and indicates to the user how much energy is available for usage. However important SOC is, it remains one of the main challenges in the successful operation of EVs [1]. SOC estimation methods are typically sophisticated, with scarce literature found to provide a detailed explanation of the many methods. Most of the methods have significant issues that become more apparent by the aging of the battery, temperature fluctuation, and change in discharge cycles [1]. Also, many of the methods produce inaccurate estimations of SOC because of the high sensitivity that lithium-ion batteries have to internal/external factors and complex electrochemical reactions [1]. This results in a model attempting to evaluate complex calculations with high computation cost, with negligence on the effects of time.

A robust number of Machine Learning (ML) algorithms, mainly comprised of Neural Network (NN) architectures, have been utilized in an attempt to accurately predict SOC due to the ability to adapt and self-learn on a complex nonlinear dataset. He et al. developed one such architecture., using a Back Propagation Neural Network (BPNN) along with an Unscented Kalman Filter (UKF) to estimate SOC during different driving conditions [2]. We present and investigation that introduces a novel approach to battery modeling by using a Recurrent Neural Network architecture that treats the battery performance data as a time-series.

2. Methods

2.1 Multi-Step Multivariate Time Series Forecasting

The input to the temporal sequence contains multiple parameters (*P)* with *N* observations $\{X_{P_1}, X_{P_2}, X_{P_3}, \dots, X_{P_N}\}$ with each data point being comprised of a real-valued vector. The target, or output, data points are tasked with predicting multi-step observations $\{\hat{y}_{P_{N+1}}, \hat{y}_{P_{N+2}}, \hat{y}_{P_{N+3}}, \dots, \hat{y}_{P_{N+H}}\}$ where; H is referred to as the forecasting *horizon*. For this investigation, eight input variables have been used; Current (A), Voltage (V), Charge Capacity (Ah), Charge Energy (W), Discharge Energy (Wh), dV/dt (V/s), and Temperature (C), along with the output of SOC.

2.2 Estimating State of Charge

When a battery is being discharged, this percentage can be expressed in the following formula with $Q_{releasable}$ as the current capacity of the system and Q_{rated} as the nominal capacity of the system:

$$SOC = \frac{Q_{releasbale}}{Q_{rated}} \times 100\% \tag{1}$$

Since batteries are comprised of chemical energy and current literature does not demonstrate any method to measure SOC directly and precisely, estimations can be broken down into four categories: 1) Direct measurement methods which measure physical properties of the battery (i.e. voltage) and estimate SOC based on linear and nonlinear relationships. 2) Book-keeping estimation that call indirect methods that utilize discharging current as the input and integrate throughout time to calculate SOC. 3) Adaptive systems that are self-designing and automatically adjust the SOC when subjected under various discharging conditions. 4) Hybrid methods that employ the advantageous parts of each SOC estimation method to provide global optimized or ensemble estimation method.

Typically, machine learning algorithms fall within the adaptive systems category. When a neural network is used for SOC prediction the model typically takes discharge current, terminal voltage and temperature as inputs and SOC as the output to build the structure of the NN for $LiFePO_4$ batteries [1]. Researchers have demonstrated that NN models have the ability of functioning in non-linear environments during battery charge/discharge conditions [1].

2.3 LSTM-RNN Architecture for SOC Prediction

Recurrent Neural Networks (RNN)

The effectiveness of a NN assumes independence of data in a training and test set [3]. Typical NN architecture for estimating SOC in literature usually include three layers; input, output, and hidden. The majority of models assume input nodes of discharge current, terminal voltage, and ambient temperature with an output of SOC to construct the model for $LiFePO_4$ [1]. Many of these models lack consideration of time and online features which limit their accuracy through the life of the battery. In an EV application, data is time-variant, that is to say that battery performance will degrade through time. Therefore, the independence assumption fails due to the data being highly affected by other data within the dataset. As a result, the learning performance of a NN will degrade through time and the capability of the predictive model will become unreliable. In the particular situation of EV operation, a data point can be highly affected by a previous data point so it becomes vital for the NN to concern itself with these dependencies. Recurrent Neural Networks (RNN) have been used extensively for applications with time-series and sequential datasets. RNN models are different than feedforward architectures [2].

Long-Short Term Memory (LSTM)

In practice, the LSTM has demonstrated a superior ability to learn long-range dependencies as compared to simplified RNNs [3]. This model was introduced in an attempt to overcome the vanishing gradients problem. While LSTM is a type of RNN, this model introduces a *memory cell*. RNNs utilize *long-term memory* in the form of weights which change slowly during training and *short-term memory* in the form of ephemeral activations which pass from one node to another [3]. The memory cell in a LSTM network has an intermediate type of storage in the form of *gates* within each hidden layer. A typical LSTM architecture contains four parts within the memory block: an input gate (*i*), a forget gate (*f*), and output gate (*o*), and cell state (C_{t-1}). The forget gate is responsible to deciding which information is retained or discarded from the cell state. The input gate determines which values will be updated to a vector of *new candidate* values $\widetilde{C}_t$. Finally, the output gate decides what information will be passed to the cell state in the next time step.

2.4 Research Focus

Based on literature, we identified specific research lines of effort 1) RNN architectures (specifically LSTM architectures) utilized for battery modeling is scarce. Literature states that battery modeling should account for battery history especially in HEV applications. 2) Battery modeling of internal dynamics is complex and requires reaching 3+ hours of steady state (in some cases) to identify an initial accurate SOC which is not ideal for HEV applications. 3) Influential parameters that highly effect battery performance are typically assumed without any feature analysis methods to backup assumptions. 4) Many of the accepted SOC prediction models are hybrid methods which require multiple mathematical models working in alignment with one another to achieve a reasonable prediction accuracy.

Therefore, an online battery model that relates physical battery properties (terminal voltage, current, temperature, historical usage) to SOC and includes operating limits is desired for design of a BMS with the main function of controlling charge/discharge of the battery and provides information to the user about future SOC estimations is the focus of this investigation.

Three research topics have been evaluated to occupy the research gaps. Research topic number 1 is Feature Engineering for determination of highly influential parameters affecting battery performance. Research topic number 2 is grid search for model tuning using high performance computing (HPC) resources. Research topic 3 is concerned with the nature of a time variant big dataset utilized for SOC prediction. The hypothesis is that during HEV application an LSTM architecture will minimize error on big data SOC prediction based on sequential data due to use of long term and short-term temporal correlations.

2.5 Model Parameters

Three input neurons were used based on physical variables (current, voltage, and temperature) collected from the battery experiment. 50 neurons make up the first hidden layer with 1 neuron in the output layer (SOC). The LSTM moves through 1-time step and contains 4 features. We performed training on 50, 100, 150, and 300 epochs with results reported in Section 4.2. The model evaluates Mean Absolute Error for the loss function, utilizes Adam optimization, and has a lag (window width) of 3. The training and test loss and Root Mean Square error (RMSE) are reported in section 4.2.

3. Experimental

3.1 Data Collection

The data was collected from a lab environment and reported in He el al. [2]. The battery tested was a $LiFePO_4$ and was subjected to three battery testing load profiles; dynamical stress testing (DST), US06 highway driving schedule, and federal urban driving schedule (FUDS). DST data was used as training due to the dataset being less complex than the other two data profiles. US06 and FUDS were used as testing because the data is complex (and highly nonlinear) and simulates real life driving conditions.

The authors used an algorithmic approach consisting of four steps. The first step initialized $k = 1, n = 1$, where k is a parameter used to determine the dimension of the input vector, and n is the number of neurons in the hidden layers. Next, the following features were trained; Current, Voltage, and Temperature as inputs and SOC as the output. Next, the root mean square error (RMSE) was calculated between the estimation output and the actual SOC. Finally, if the RMSE <1%, the searing is completed; else if $k < 2n, k = k + 1$ and return to step two; else $n = n + 1, k = n$ and return to step two.

3.2 Data Formatting

The experimentation from He et al. resulted in the collection of a very large multivariate dataset. For this investigation, we prepared, cleaned, and organized the dataset to allow for quality results. We eliminated empty rows and kept columns containing current, voltage, temperature, and SOC values. The results later discussed in this article pertain only to a small fraction of the entire dataset. One test run from each load profile was selected for evaluation in the LSTM model (DST, US06, FUDS).

Once cleaning was complete, the time series data was transformed into a supervised learning format. Using a lag of 3, this altered the data from $\{var1, var2, \dots varP\}$ to
$\{var1_{(t-3)}, var1_{(t-2)}, var1_{(t-1)}, var1_{(t)}, var1_{(t+1)}, var1_{(t+2)}, var1_{(t+3)}, \dots,$

$varP_{(t-3)}, varP_{(t-2)}, varP, varP_{(t)}, varP_{(t+1)}, varP_{(t+2)}, varP_{(t+3)}\}$. Finally, the data was normalized to a value from 0 to 1.

4. Results and Discussion

4.1 Feature Analysis

A main aspect of this investigation was to determine the effects of physical battery parameters on battery operation. Table 1 reports the importance values for the physical properties collected during experimentation. We see that current and voltage are the most influential parameters during battery operation, which was expected. The remaining parameters have negligent influence on how the battery performs.

Correlation analysis was performed on the DST dataset and results are reported in Figure 1. Only one sample was selected to feed the LSTM however, for comparison purposes three samples were arbitrarily selected. Looking at the correlation plots (shown in Figure 1), we can observe that voltage and temperature are not correlated with SOC.

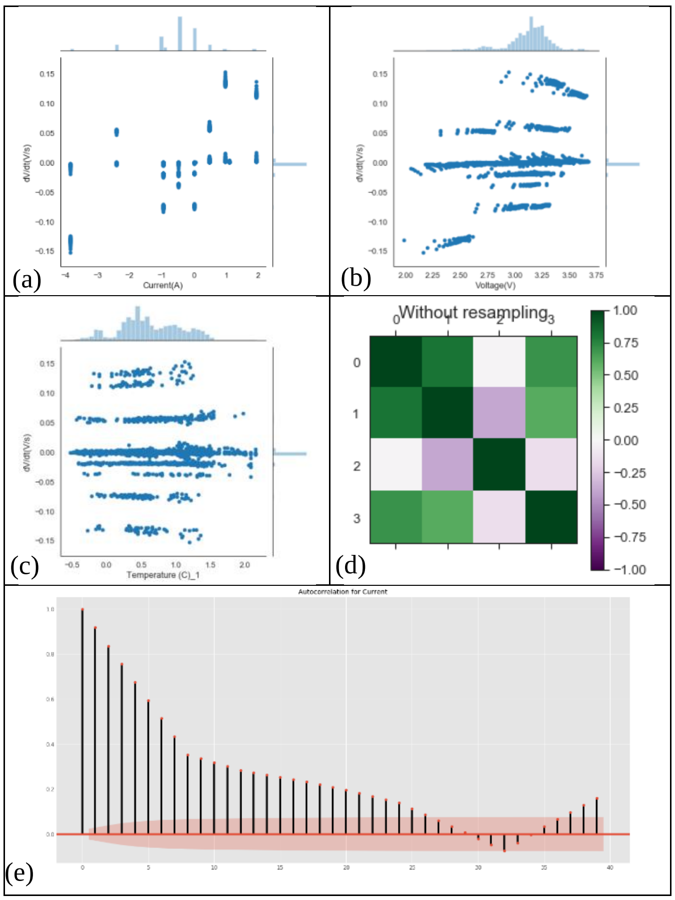

Figure 1: Correlation plots of (a) Current and SOC (b) Voltage and SOC (c) Temperature and SOC (d) Correlation among all parameters used for the LSTM input layer and (e) Autocorrelation of the current feature over time in the DST dataset

Table 1: Feature Analysis using Random Forest Techniques

Feature	Importance Value
Current (A)	0.61
Voltage (V)	0.2
Discharge Capacity (Ah)	0.07
Discharge Energy (Wh)	0.07
Charge Capacity (Ah)	0.03
Charge Energy (Wh)	0.03

Table 2: RMSE Results

	Number of Epochs	RMSE
DST	50	0.379
	100	0.376
	150	**0.372**
	300	0.373
US06	50	0.528
	100	0.518
	150	0.513
	300	0.464
FUDS	50	**0.534**
	100	0.509
	150	0.503
	300	0.493

4.2 LSTM Results

We set the lag for the LSTM to 3 with a variety of epochs executed. Table 2 shows the result of RMSE by four different epochs (50, 100, 150 and 300) and compares the RMSE among the different data sets. The highest RMSE was recorded on the most complex dataset with the least number of epochs, while the lowest RMSE was recorded on the simplest dataset with epochs ≥ 150. The best recorded RMSE, 0.372, was observed in the DST dataset while running 150 epochs. It's interesting to see that RMSE becomes steady between 150 and 300 epochs as the RMSE only changes 0.001. This result suggests that additional epochs are not needed for simple time series datasets for battery operation. The highest RMSE, 0.534, was recorded with the FUDS dataset with 50 epochs.

From the He et al. results, their NN algorithm RMSE was 1.5 to 4.1 RMSE for the US06 database and 1.7 to 4.2 for the FUDS database. For the NN+UKF algorithm, the RMSE for the US06 was 0.9 to 2.5 and 0.5 to 2.2 for the FUDS. It is important to note that the authors report results of the NN with and without a UKF. UKF was added to smooth the higher error % from the NN battery model. The LSTM in this investigation outperforms the measured performance from the NN+UKF without needing a filter (see Table 2).

The loss of the train and test sets can be observed in Figure 2. Results for 50 epochs (left column) are reported for the DST (top), US06 (middle), and FUDS (bottom) datasets and compared to results for 300 epochs (right column).

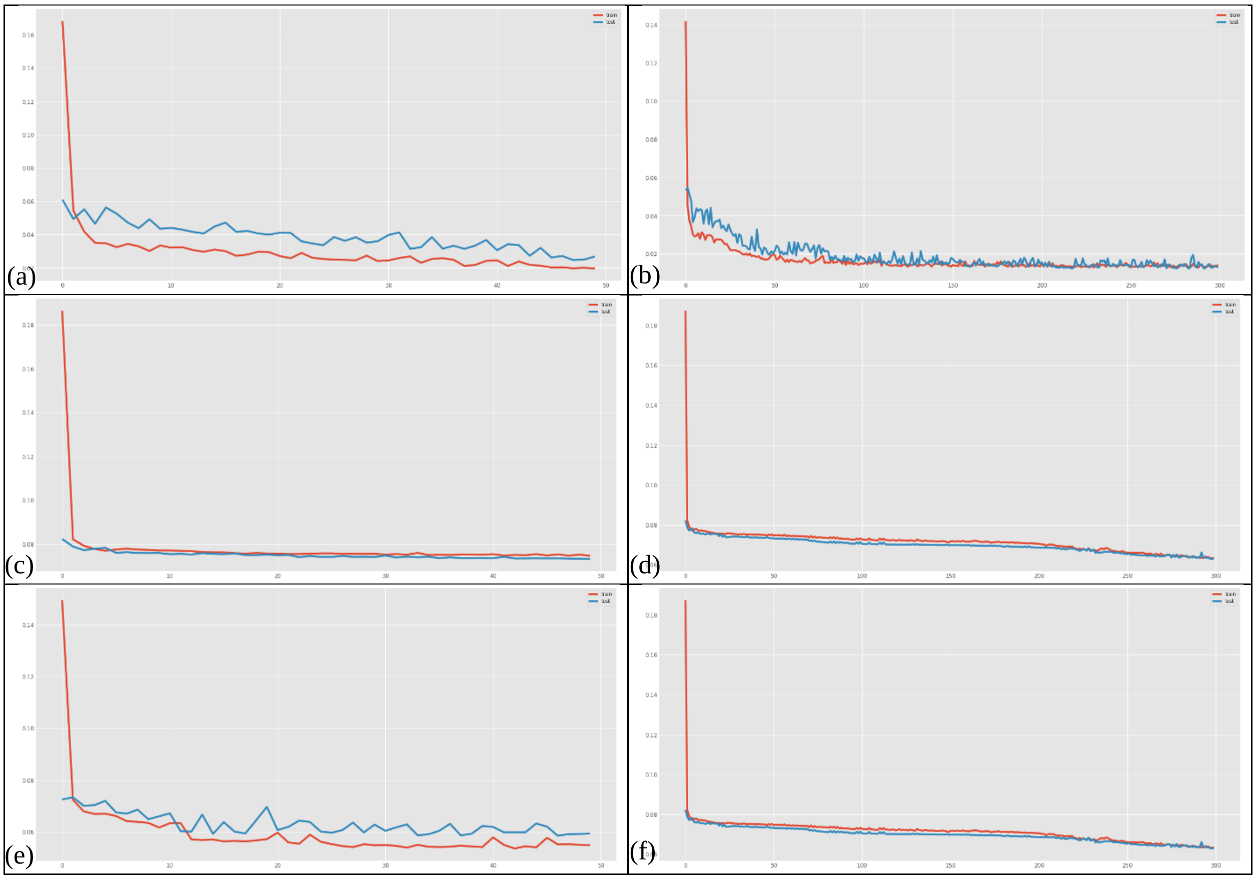

Figure 2: Loss for train vs test sets for (a) 50 epochs with DST dataset (b) 300 epochs with DST dataset (c) 50 epochs with US06 dataset (d) 300 epochs with US06 dataset (e) 50 epochs with FUDS dataset and (f) 300 epochs with FUDS dataset

The experimental data points are compared to the predicted data points in Figure 3, confirming the reported RMSE results. It is clear that the predicted values mirror the actual values. Figure 3 shows the plots for actual and predicted values for 50 epochs (left column) and 300 epochs (right column) for each load profile dataset; DST (top), US06 (middle) and FUDS (bottom).

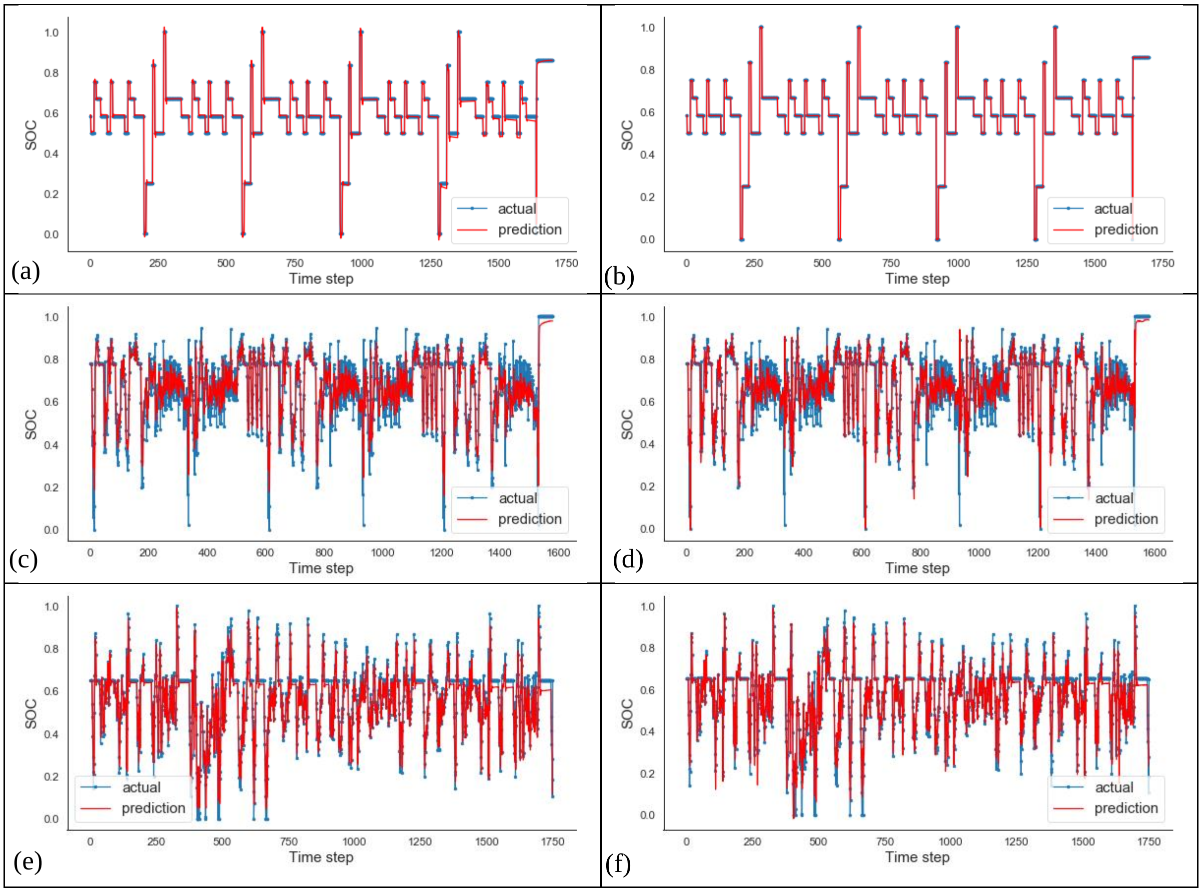

Figure 3: Actual vs Predicted values for (a) 50 epochs with DST dataset (b) 300 epochs with DST dataset (c) 50 epochs with US06 dataset (d) 300 epochs with US06 dataset (e) 50 epochs with FUDS dataset and (f) 300 epochs with FUDS dataset

5. Conclusion

This investigation was focused on creation and implementation of a Recurrent Neural Network using temporal multivariate data and ambient temperature features. Several objectives remain for this investigation. Most importantly is the implementation of High-Performance Computing (HPC). The dataset is large and evaluating the entire dataset will require the use of HPC. A grid search will be performed and hyperparameters will be tuned for the model. These hyperparameters will be used to optimize the LSTM model.

Acknowledgements

We would like to thank the Center for Advanced Life Cycle Engineering (CALCE), University of Maryland for kindly sharing data and insights for this reliability and prediction study.

References

[1] M. Hannan, M. Lipu, A. Hussain, and A. Mohamed, “A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: Challenges and recommendations,” *Renewable and Sustainable Energy Reviews*, vol. 78, pp. 834–854, 2017.

[2] W. He, N. Williard, C. Chen, and M. Pecht, “State of charge estimation for Li-ion batteries using neural network modeling and unscented Kalman filter-based error cancellation,” *International Journal of Electrical Power & Energy Systems*, vol. 62, pp. 783–791, 2014.

[3] Z.C. Lipton, J. Berkowitz, and C. Elkan. "A critical review of recurrent neural networks for sequence learning." *arXiv preprint arXiv:1506.00019*, 2015.