On-Board Prediction of Remaining Useful Life of Lithium-Ion Battery

Final Report
January 2019

Sponsored by
Midwest Transportation Center

U.S. Department of Transportation
Office of the Assistant Secretary for
Research and Technology

IOWA STATE UNIVERSITY
Institute for Transportation

About MTC

The Midwest Transportation Center (MTC) is a regional University Transportation Center (UTC) sponsored by the U.S. Department of Transportation Office of the Assistant Secretary for Research and Technology (USDOT/OST-R). The mission of the UTC program is to advance U.S. technology and expertise in the many disciplines comprising transportation through the mechanisms of education, research, and technology transfer at university-based centers of excellence. Iowa State University, through its Institute for Transportation (InTrans), is the MTC lead institution.

About InTrans

The mission of the Institute for Transportation (InTrans) at Iowa State University is to develop and implement innovative methods, materials, and technologies for improving transportation efficiency, safety, reliability, and sustainability while improving the learning environment of students, faculty, and staff in transportation-related fields.

ISU Non-Discrimination Statement

Iowa State University does not discriminate on the basis of race, color, age, ethnicity, religion, national origin, pregnancy, sexual orientation, gender identity, genetic information, sex, marital status, disability, or status as a U.S. veteran. Inquiries regarding non-discrimination policies may be directed to Office of Equal Opportunity, 3410 Beardshear Hall, 515 Morrill Road, Ames, Iowa 50011, Tel. 515-294-7612, Hotline: 515-294-1222, email eooffice@iastate.edu.

Notice

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. The opinions, findings and conclusions expressed in this publication are those of the authors and not necessarily those of the sponsors.

This document is disseminated under the sponsorship of the U.S. DOT UTC program in the interest of information exchange. The U.S. Government assumes no liability for the use of the information contained in this document. This report does not constitute a standard, specification, or regulation.

The U.S. Government does not endorse products or manufacturers. If trademarks or manufacturers' names appear in this report, it is only because they are considered essential to the objective of the document.

Quality Assurance Statement

The Federal Highway Administration (FHWA) provides high-quality information to serve Government, industry, and the public in a manner that promotes public understanding. Standards and policies are used to ensure and maximize the quality, objectivity, utility, and integrity of its information. The FHWA periodically reviews quality issues and adjusts its programs and processes to ensure continuous quality improvement.

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle	5. Report Date
On-Board Prediction of Remaining Useful Life of Lithium-Ion Battery	January 2019
	6. Performing Organization Code

7. Authors	8. Performing Organization Report No.
Chao Hu, Sheng Shen, and Yifei Li	

9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
Iowa State University Department of Mechanical Engineering 1138 Pearson Hall Ames, IA 50011	**11. Contract or Grant No.** Part of DTRT13-G-UTC37

12. Sponsoring Organization Name and Address		13. Type of Report and Period Covered
Midwest Transportation Center 2711 S. Loop Drive, Suite 4700 Ames, IA 50010-8664 Iowa State University 1138 Pearson Hall Ames, IA 50011	U.S. Department of Transportation Office of the Assistant Secretary for Research and Technology 1200 New Jersey Avenue, SE Washington, DC 20590	Final Report **14. Sponsoring Agency Code**

15. Supplementary Notes

Visit www.intrans.iastate.edu for color pdfs of this and other research reports.

16. Abstract

This project was intended to create an intelligent prognostics platform for lithium-ion (Li-ion) batteries, which would equip existing battery management systems with the capability to perform predictive maintenance/control for failure prevention. The platform developed in this project consisted of two modules:

- Deep feature learning, which automatically learns the features of (capacity) fade from large volumes of voltage and current measurement data during partial charge cycles and estimates the real-time state of health (SOH) of a battery cell in operation
- Ensemble prognostics, which leverage the current and past SOH estimates in Module 1 to achieve robust prediction of the cell's remaining useful life

Robust prediction of remaining useful life was achieved by ensemble learning-based prognostics, which synthesized the generalization strengths of multiple prognostic algorithms to ensure high prediction accuracy for an expanded range of battery applications and their operating conditions. The two modules aimed to learn features of fade from partial charge data, assess real-time health of individual battery cells, and predict when and how the cells are likely to fail. A case study involving implantable-grade Li-ion cells was conducted to demonstrate a deep learning approach to online capacity estimation, developed for Module 1.

17. Key Words	18. Distribution Statement
battery management system—deep learning—lithium-ion battery—remaining useful life—state of health	No restrictions.

19. Security Classification (of this report)	20. Security Classification (of this page)	21. No. of Pages	22. Price
Unclassified.	Unclassified.	35	NA

Form DOT F 1700.7 (8-72) **Reproduction of completed page authorized**

ON-BOARD PREDICTION OF REMAINING USEFUL LIFE OF LITHIUM-ION BATTERY

Final Report
January 2019

Principal Investigator
Chao Hu, Assistant Professor
Mechanical Engineering, Iowa State University

Research Assistants
Sheng Shen and Yifei Li

Authors
Chao Hu, Sheng Shen, and Yifei Li

Sponsored by
Iowa State University,
Midwest Transportation Center, and
U.S. Department of Transportation
Office of the Assistant Secretary for Research and Technology

A report from
Institute for Transportation
Iowa State University
2711 South Loop Drive, Suite 4700
Ames, IA 50010-8664
Phone: 515-294-8103 / Fax: 515-294-0467
www.intrans.iastate.edu

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

The authors would like to thank the Midwest Transportation Center and the U.S. Department of Transportation Office of the Assistant Secretary for Research and Technology for sponsoring this research. The authors are also grateful to Iowa State University for providing the required match funds for this study.

A previous version of this report was published as part of a conference proceeding and a journal article:

Shen, S., M. K. Sadoughi, X. Chen, M. Hong, and C. Hu. 2018. Online Estimation of Lithium-Ion Battery Capacity Using Deep Convolutional Neural Networks. *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE)*, Paper Number: DETC2018-86347, August 26–29 2018, Quebec City, Quebec, Canada. http://proceedings.asmedigitalcollection.asme.org/proceeding.aspx?articleid=2713204.

Li, Z., D. Wu, C. Hu, and J. Terpenny. 2019. An Ensemble Learning-Based Prognostic Approach with Degradation-Dependent Weights for Remaining Useful Life Prediction. *Reliability Engineering and System Safety*, Vol. 184, pp. 110–122. https://www.sciencedirect.com/science/article/pii/S0951832017308104.

EXECUTIVE SUMMARY

Lithium-ion (Li-ion) battery technology has played a critical role in supporting wide-scale adoption of hybrid and electric vehicles (HEVs/EVs). Real-time health diagnostics/prognostics and predictive maintenance/control of Li-ion batteries are essential for reliable and safe battery operation. Over the past two decades, a number of machine learning techniques have been developed to estimate the state of health (SOH) of a battery cell based on readily available measurements from the cell (i.e., voltage, current, and temperature). However, most of these existing techniques required manual feature extraction that relied heavily on engineering skills and domain expertise, and may risk dropping useful information in the raw data that would otherwise help improve the diagnostic performance. Additionally, existing data-driven and model-based prognostics approaches are primarily application-specific and thus are not easily generalized for prognostics of Li-ion batteries used in different applications.

This project intended to create an intelligent prognostics platform for Li-ion batteries, which would equip existing battery management systems (BMSs) with the capability to perform predictive maintenance/control for failure prevention. The platform developed in this project consisted of two modules:

- Deep feature learning, which automatically learns features of (capacity) fade from large volumes of voltage and current measurement data during partial charge cycles and estimates the real-time SOH of a battery cell in operation

- Ensemble prognostics, which leverage the current and past SOH estimates in Module 1 to achieve robust prediction of the cell's remaining useful life (RUL)

Robust prediction of RUL is achieved by ensemble learning-based prognostics, which synthesizes the generalization strengths of multiple prognostic algorithms to ensure high prediction accuracy for an expanded range of battery applications and their operating conditions. The two modules aimed to learn features of fade from partial charge data, assess real-time health of individual battery cells, and predict when and how the cells are likely to fail. A case study involving implantable-grade Li-ion cells was conducted to demonstrate a deep learning approach to online capacity estimation, developed for Module 1.

INTRODUCTION

Lithium-ion (Li-ion) battery technology has been playing a critical role in supporting wide-scale adoption of hybrid and electric vehicles (HEVs/EVs). Safety and reliability of the Li-ion batteries in these transportation applications have been receiving a considerable amount of attention from both industry and academia. Over the past two decades, real-time health diagnostic/prognostic techniques have been developed to monitor the state of health (SOH) and predict the remaining useful lives (RULs) of individual battery cells (Plett 2004, Lu et al. 2013). These techniques are often deployed in battery management system (BMS) to perform the following two on-board tasks:

- Health diagnostics: The purpose of this task is to estimate the capacity of a battery cell based on readily available measurements (i.e., voltage [*V*], current [*I*], and temperature [*T*]) from the cell. Here, the cell's capacity is the total amount of charge stored in the cell when the cell is fully charged. It is an important indicator of the cell SOH.

- Health prognostics: The purpose here is to predict the remaining service time or number of charge-discharge cycles before the cell capacity fades to an end of life limit. The remaining time or number of charges is also termed the RUL.

The remainder of this section first reviews the state of knowledge on battery health diagnostics/prognostics and then provides an overview of the proposed intelligent prognostics platform.

Review on Battery Health Diagnostics

Two major parameters characterizing the SOH of a Li-ion battery cell are the capacity and internal resistance of the cell (Lu et al. 2013). When the capacity of a cell is used to indicate its SOH, the cell SOH is often defined as the ratio of the cell capacity at the current charge-discharge cycle to the rated capacity given by the cell manufacturer. Accurate capacity estimation allows a battery cell to be replaced right before the cell capacity fades to an end of life limit, thus allowing cell useful life to be fully exploited while preventing imminent cell failures.

Recent literature has reported a large number of approaches to estimating the capacity of a Li-ion battery cell. These approaches can be broadly categorized into (1) adaptive filtering approaches (Plett 2004, Plett 2006, Lee et al. 2008, Schmidt et al. 2010, Hu et al. 2012a, Xiong et al. 2014, Chen et al. 2018), (2) coulomb counting approaches (Ng et al. 2009, Waag and Sauer, 2013), and (3) machine learning-based approaches (Eddahech et al. 2012, Kim et al. 2012, Nuhic et al. 2013, Bai et al. 2014, Hu et al. 2015, Liu et al. 2018). Of particular interest here are the machine learning-based approaches that perform online capacity estimation by learning the complex dependency of the capacity of a cell on the characteristic features extracted from the *V* and *I* measurement data from the cell. For example, Hu et al. (2015) first defined several characteristic features, indicative of the cell capacity, based on the partial charge curves and then employed the relevance vector machine, a kernel regression model, to determine the relationship between the

capacity of a battery cell and its charge-related features. The regression model, after being trained offline, was used online to infer the unknown capacity of a battery cell from the predefined set of charge-related features (Hu et al. 2015). This category of capacity estimation approaches has gained popularity in recent years, largely due to the increased capability to collect large volumes of measurement data and leverage the data to train complex machine learning models.

Although the existing machine learning-based approaches can provide the satisfactory accuracy in capacity estimation, they require manually extracting (or defining) a certain number of features from the voltage and current measurements. It may be difficult to manually identify the characteristic features that carry the most useful information for estimating the cell capacity. In addition, extracting a limited number of features may risk the loss of critical information in the raw data that would otherwise help improve the diagnostic performance. Therefore, it becomes essential to automate the feature extraction process, thus allowing for learning the informative features from the raw data

Review on Battery Health Prognostics

Health prognostics is the process of predicting the RUL of an engineered system. RUL refers to the available service time left before the performance of the system degrades to an unacceptable level. Research on health prognostics of a general engineered system has been conducted with an emphasis on modeling the RUL distribution. In general, three categories of approaches have been developed that enable continuous updating of system health degradation and RUL distribution: (1) model-based approaches (Gebraeel et al. 2005, Luo et al. 2008, Gebraeel and Pan 2008, Wang et al. 2014), (2) data-driven approaches (Wang et al. 2008, Heimes 2008, Coble and Hines 2008, Zio and Di Maio 2010, Hu et al. 2012b), and (3) hybrid approaches (Goebel et al. 2006, Saha et al. 2009, Hu et al. 2012b, Liao and Köttig 2014, Liao and Köttig 2016). Model-based prognostics refers to the methods that use models derived from first principles or probability theory. The prediction accuracy of model-based approaches depends on the prior knowledge of physical behavior (Liao and Köttig 2014). However, the domain knowledge for complex systems is not always available or may be too expensive to acquire. To complement model-based approaches, data-driven approaches refer to the approaches that use models learned exclusively from data. Typical data-driven methods use interpolation (Wang et al. 2008, Zio and Di Maio 2010), extrapolation (Coble and Hines 2008), and machine learning (Heimes 2008) for RUL prediction. In data-driven approaches, training data are used to design and train a predictive model; testing data are used to validate the predictive model. Data-driven approaches are typically more effective than model-based approaches for complex engineered systems. Hybrid prognostics refers to the approaches that facilitate the combined use of model-based and data-driven approaches.

Battery prognostics enable an on-board BMS to predict the remaining service time or number of charge-discharge cycles before the capacity of a battery cell managed by the BMS fades below an end of life limit. Research on battery health prognostics has been conducted mainly by researchers in the Prognostics and Health Management (PHM) Society. Note that the approaches mentioned earlier, although not developed specifically for Li-ion battery prognostics, can

generally be adapted for RUL assessment of Li-ion batteries. One of the earliest studies on Li-ion battery prognostics proposed a Bayesian framework with a particle filter for RUL prediction of individual battery cells based on impedance measurement (Saha et al. 2009). To eliminate the need for impedance measurement equipment, researchers developed various model-based approaches that predicted RUL by extrapolating a capacity fade model (Saha and Goebel 2009, Liu et al. 2010, He et al. 2011, Miao et al. 2013, Wang et al. 2013, Ng et al. 2014, Wang et al. 2016, Hu et al. 2018).

Existing data-driven and model-based prognostic approaches are mostly application-specific, and thus cannot be readily generalized for prognostics of Li-ion batteries used in different applications. Hybrid prognostics show great potential for achieving better prediction accuracy and robustness than model-based and data-driven prognostics. Ensemble learning-based prognostics (or ensemble prognostics) are among the most popular hybrid approaches, and have been shown to be capable of improving prediction accuracy by combining multiple learning algorithms (Hu et al. 2012b). To fully harness the benefits offered by ensemble learning, the following issue needs to be addressed: How can the effects of time-dependent degradation be taken into account when predicting the RUL of an engineered system? This presents a unique opportunity to develop a new ensemble learning-based approach for prognostics of Li-ion batteries as well as general engineered systems.

Overview of Intelligent Prognostics Platform

This project was intended to create an intelligent prognostics platform for Li-ion batteries, which would equip existing battery management systems (BMSs) with the capability to perform predictive maintenance/control for failure prevention. As shown in Figure 1, the platform developed in this project consisted of two modules:

- Deep feature learning, which automatically learns features of (capacity) fade from large volumes of voltage and current measurement data during partial charge cycles and estimates the real-time SOH of a battery cell in operation

- Ensemble prognostics, which leverage the current and past SOH estimates in Module 1 to achieve robust prediction of the cell's remaining useful life (RUL)

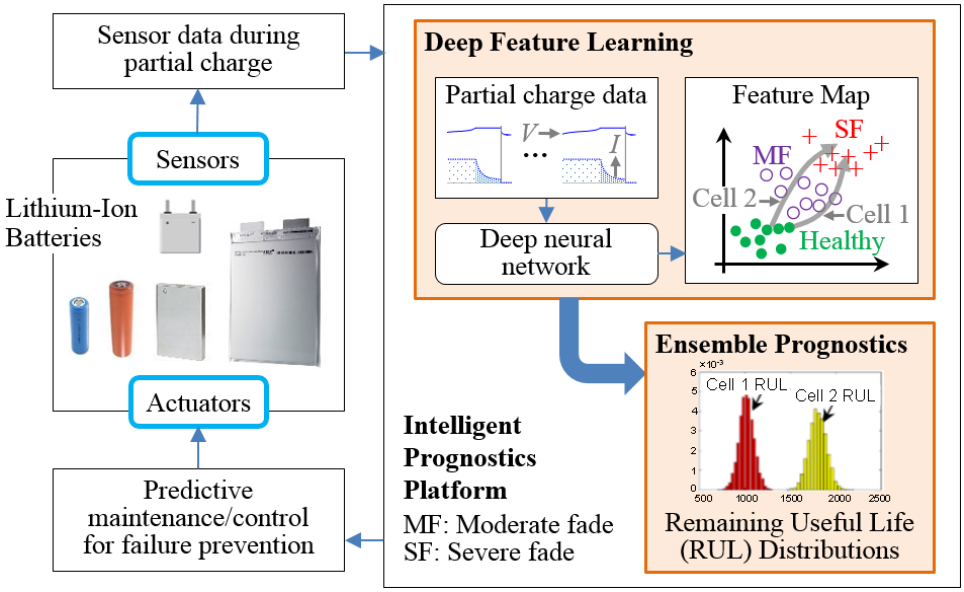

Figure 1. Intelligent prognostics platform for predictive maintenance/control of Li-ion batteries

Robust prediction of RUL is achieved by ensemble learning-based prognostics, which synthesize the generalization strengths of multiple prognostic algorithms to ensure high prediction accuracy for an expanded range of battery applications and their operating conditions. The two modules aim to learn features of fade from partial charge data, assess real-time health of individual battery cells, and predict when and how the cells are likely to fail. A case study involving implantable-grade Li-ion cells was conducted to demonstrate a new deep learning approach to online capacity estimation, developed for Module 1.

MODULE 1: DEEP FEATURE LEARNING

As discussed in the previous section, there has been a considerable amount of research conducted to develop machine learning-based approaches to online capacity estimation; however, most of these existing approaches require manual feature extraction, where human engineers manually identify the informative features from the measurement data. There is a critical need to develop a generic learning approach to automate the feature extraction process. This section presents the use of a deep learning model, namely deep convolutional neural networks (DCNNs), in Module 1 to assess the capacity of a battery cell online, based on the voltage and current measurements during a partial charge cycle.

Input and Output Structures

The objective of this study was to estimate the capacity of a Li-ion cell based on the input curves (i.e., voltage, current, and capacity curves) of the cell during a partial charge cycle. It should be noted that the voltage and current can be directly measured from the cell, while the capacity curve needs to be calculated using the coulomb counting method, which integrates the charge current over time for the partial charge cycle. An illustration of the three input curves of a cell for one charge cycle is shown in Figure 2.

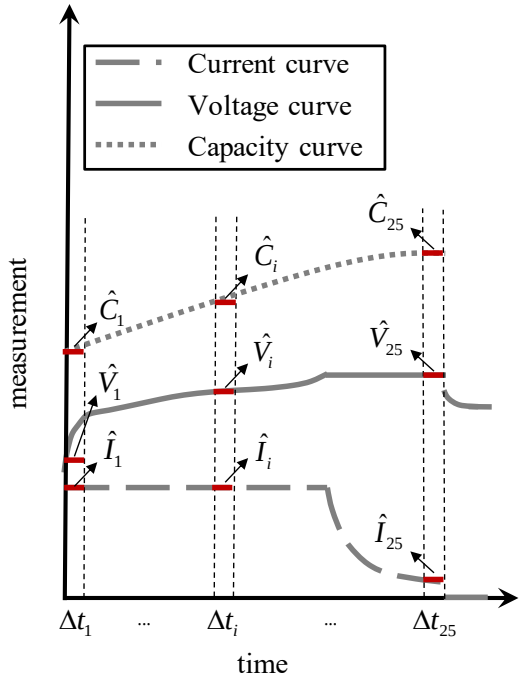

Figure 2. Voltage, current, and capacity curves of a battery cell for one charge cycle

Each curve is discretized into 25 segments (corresponding to 25 equal time steps) and the discretized values of voltage, current, and capacity are considered as the input variables to be fed

into the DCNNs model. As such, the input to the model is a 25×3 matrix, of which the first, second, and third columns correspond to the discretized values of voltage, current, and capacity, respectively. The matrix can be expressed as:

$$\text{Input} = \begin{bmatrix} \hat{V}_1 & \hat{I}_1 & \hat{C}_1 \\ \vdots & \vdots & \vdots \\ \hat{V}_i & \hat{I}_i & \hat{C}_i \\ \vdots & \vdots & \vdots \\ \hat{V}_{25} & \hat{I}_{25} & \hat{C}_{25} \end{bmatrix}_{25\times 3} \quad (1)$$

where $\hat{V}_i, \hat{I}_i$, and $\hat{C}_i$ are the discretized values of voltage, current, and capacity for the i^{th} segment, measured at the i^{th} time step (Δt_i) of a partial charge cycle. Each partial charge cycle is associated with a measured discharge capacity which is a scalar number and serves as the true output of the DCNNs model. The discharge capacity is calculated using the coulomb counting method, which integrates the discharge current over time for the entire full discharge cycle that immediately follows the partial charge cycle.

Overall Architecture of DCNNs

This study investigated the use of a deep architecture, DCNNs, to online assess the capacity of a battery cell by solving this high-dimensional mapping problem (i.e., $R^{75} \rightarrow R^1$). The DCNNs are mainly composed of two types of layers: convolutional layers (Conv.) and fully-connected layers (FC.). Convolutional layers are used to execute a special kind of linear operation named convolution. In the context of deep learning, convolution is an operation on the inputs and kernels. Specifically, each unit of a convolutional layer is connected to local patches in the feature maps of the previous layer through a set of weights called filter banks. The result of this locally weighted sum is then passed through a variety of layers, such as a rectified linear units (ReLU) (Nair and Hinton 2010) and batch normalization (BN) (Ioffe and Szegedy 2015), to form the feature maps of the next layer. Fully-connected layers use matrix multiplication by a matrix of parameters with a separate parameter describing the pair-wise interactions between all the input and output units.

The overall architecture of DCNNs, as depicted in Figure 3, is composed of eight building blocks, the first five being convolution stages and the remaining three being fully-connected stages.

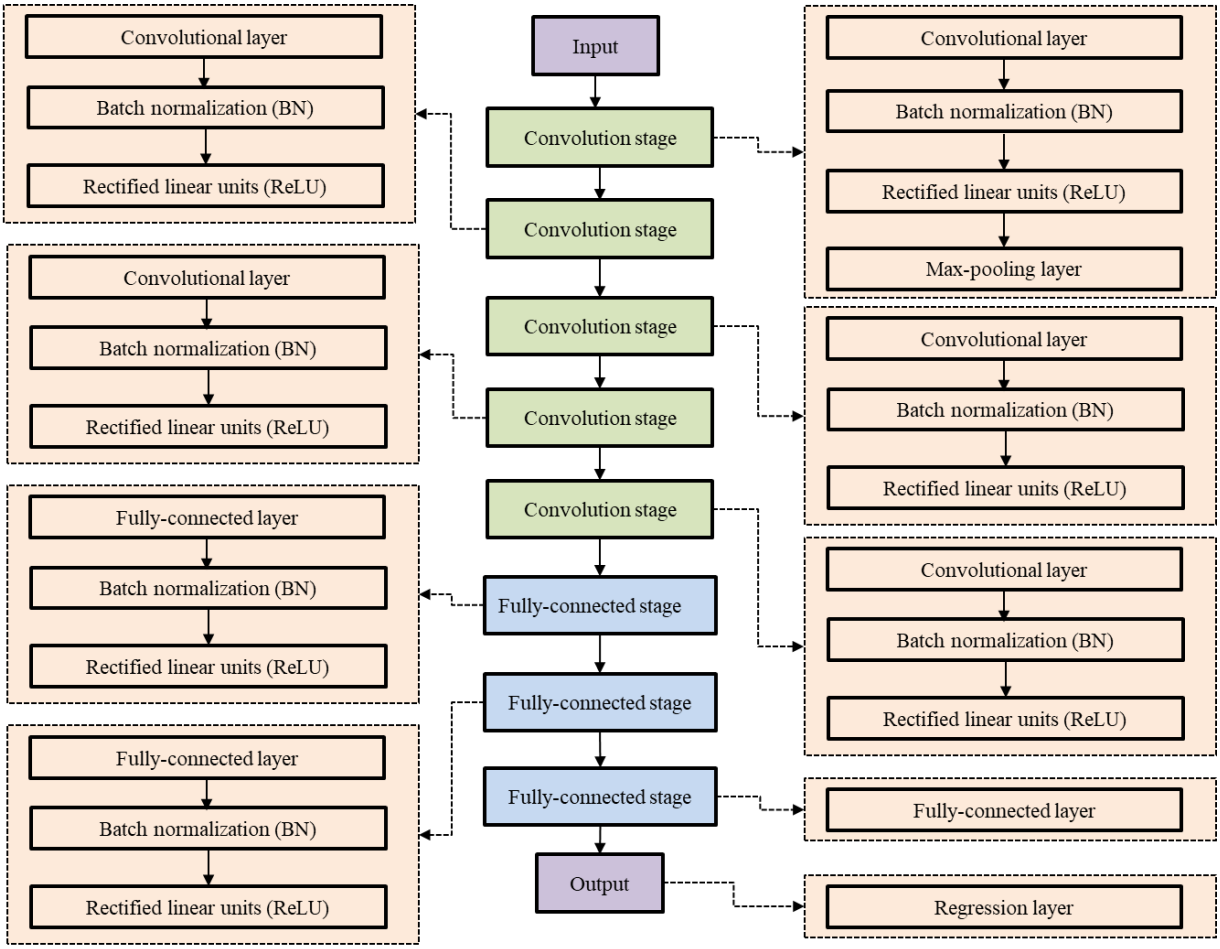

Figure 3. Architecture of the proposed DCNNs model in this study

In this study, the networks have 28,000 parameters and 5,841 neurons, and consist of five convolutional layers, one of which is followed by a max-pooling layer and three fully connected layers with a regression layer. BN is applied to the outputs of all convolutional and fully-connected layers in the proposed DCNNs. Following BN, ReLU is used to introduce nonlinearity to the networks. The max-pooling layer is only applied to the output of ReLU at the first convolutional layer. The output of the last fully-connected layer is fed into a regression layer to predict the target output that is a continuous variable.

As discussed in the previous subsection, the inputs to the DCNNs consisted of a sample matrix with fixed size 25×3×1 (i.e., the height, width, and number of channels of the sample matrix). Thus, the networks' inputs consisted of 75 measurements (i.e., F_1^1, F_1^2, …, F_{25}^3) that can be divided into three groups: 1) voltage $\mathbf{F}^1 = \{F_1^1, F_2^1, \ldots, F_{25}^1\}$, 2) current $\mathbf{F}^2 = \{F_1^2, F_2^2, \ldots, F_{25}^2\}$, and 3) charge capacity $\mathbf{F}^3 = \{F_1^3, F_2^3, \ldots, F_{25}^3\}$. These measurements represented the entire dataset collected from a cell during a partial charge cycle. In the first convolution stage, the convolutional layer filtered the input samples with 16 kernels (also called filters) of size $1 \times 2 \times 1$ with a stride of size 1×1. The filter moved along the input sample matrix vertically and

horizontally, repeating the same computation for each region, that is, convolving the input. The step size with which the filter moved is called the stride. The samples, in turn, were fed into BN, ReLU, and the max-pooling layer of this stage. The second convolution stage received the outputs from the first convolution stage and fed it into the convolutional layer of this stage, and filtered it with 32 kernels of size $3 \times 1 \times 1$. The third, fourth, and fifth convolution stages are connected sequentially by taking the responses from the ReLU of the previous stage as the inputs to the next stages. They all possessed 40 kernels of size $3 \times 1 \times 1$ with a stride of size 1×1. The fully connected layers had 40 neurons, each of which connected to all neurons in the previous layer. After passing through a stack of convolution stages and fully connected stages, the input sample reached the destination, the regression layer, which outputted an estimated target corresponding to the input sample.

Training Algorithm

Like any other machine learning method, DCNNs contain a set of unknown parameters that need to be identified using a training dataset. To this end, a cost function *J* was defined to quantify the differences between the DCNNs' predictions and the true outcomes of the training dataset. The stochastic gradient descent (SGD) method was utilized to minimize the expected generalization error given by the cost function. SGD is probably the most widely used optimization algorithm in deep learning, particularly when the training process is very slow due to a large number of available data. In this study, SGD with momentum, originally designed by B. T. Polyak (1964), was employed as the optimization algorithm to accelerate the learning process. SGD with momentum updated the parameters, $\boldsymbol{\theta}$ (weights, $\boldsymbol{\omega}$, and biases, $\mathbf{b}$), to minimize the generalization error by taking small steps in the direction of the negative gradient of the cost function. In other words, SGD with momentum accumulated an exponentially decaying moving average of the past gradients and continued to move in their direction. The cost function $J(\boldsymbol{\theta})$ with the regularization term (also called weight decay) can be expressed as:

$$J_R(\boldsymbol{\theta}) = J(\boldsymbol{\theta}) + \lambda\Omega(\boldsymbol{\omega}) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_{\boldsymbol{\theta}}\left(\mathbf{x}^{(i)}\right) - z^{(i)}\right)^2 + \frac{\lambda}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} \tag{2}$$

where $J_R(\boldsymbol{\theta})$ is the cost function with the regularization term $\lambda\Omega(\boldsymbol{\omega})$, λ is the L_2 regularization factor that weighs the relative contribution of the norm penalty term, $\Omega(\boldsymbol{\omega})$, $h_{\boldsymbol{\theta}}(\mathbf{x})$ is the hypothesis function, m is the number of training samples used at each iteration, $\mathbf{x}^{(i)}$ is the feature vector of the i^{th} training sample, and $z^{(i)}$ is the associated target value. The hypothesis function $h_{\boldsymbol{\theta}}(\mathbf{x})$ is defined as:

$$h_{\boldsymbol{\theta}}(\mathbf{x}) = b_0x_0 + \omega_1x_1 + \cdots + \omega_nx_n \tag{3}$$

where *n* is the input dimension, and x_n and ω_n denote the n^{th} input variable and its parameter, respectively ($x_0 = 1$, and b_0 is the bias value). Then, the parameter $\boldsymbol{\theta}$ can be iteratively updated according to the following:

$$\hat{g} = \frac{1}{2q}\sum_{i=1}^{q}\left(h_\theta\left(\mathbf{x}_j^{(i)}\right) - z_j^{(i)}\right)^2 \tag{4}$$

$$\boldsymbol{\theta}_{j+1} = \boldsymbol{\theta}_j - \alpha\hat{g} + \gamma\left(\boldsymbol{\theta}_j - \boldsymbol{\theta}_{j-1}\right) - \lambda\alpha\boldsymbol{\theta}_j \quad (5)$$

where $\hat{g}$ is an estimator of the exact gradient by sampling a minibatch of q samples, $\mathbf{x}_j^{(i)}$ is the feature vector of the i^{th} training sample among the minibatch at the j^{th} iteration, and $z_j^{(i)}$ is the corresponding target value, α is the initial learning rate (or step size), γ is the momentum that determines the contribution of gradients from the previous iteration to the current iteration, and $\boldsymbol{\theta}_j$ denotes the parameter estimate at the j^{th} iteration.

The weights in each layer were randomly initialized according to a Gaussian distribution with the mean of 0 and standard deviation of 0.01, while the values of biases of all convolutional and fully connected layers were initialized at 0.

MODULE 2: ENSEMBLE PROGNOSTICS

As discussed in the Introduction section, robust RUL predictions can be achieved by forming an ensemble of multiple individual prognostic algorithms. This study deviated from the existing ensemble prognostics approach (Hu et al. 2012b) that identified degradation-independent weights for member algorithms. Instead, this study created a new ensemble approach that used degradation-dependent weights to combine multiple member algorithms in a weighted-sum formulation. This section presents the new ensemble learning-based prognostic approach. Although the new approach was developed in the context of general prognostics, it is expected that the methodology is readily applicable to RUL assessment of Li-ion batteries.

A Generic Computational Framework

A generic computational framework of the ensemble learning-based prognostic approach with degradation-dependent weights is illustrated in Figure 4.

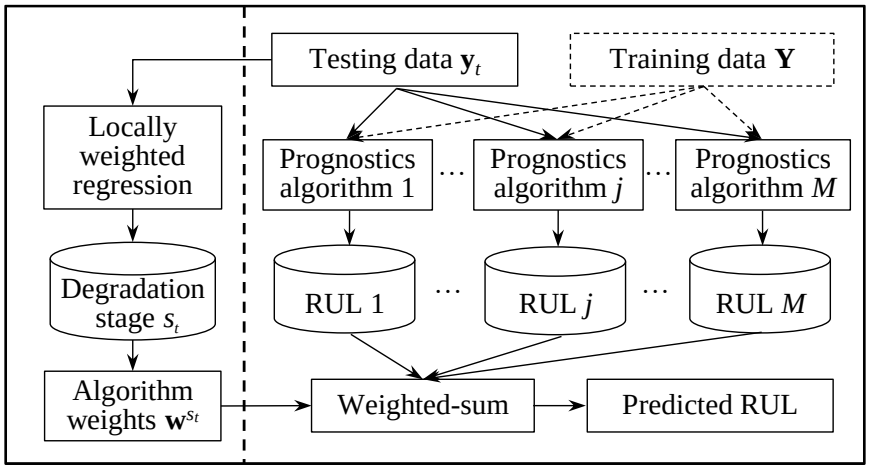

Figure 4. Flowchart of the proposed approach for ensemble prognostics

A training dataset $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N]^T$ includes multi-dimensional measurement data from N different run-to-failure units, where $\mathbf{y}_i$ (i = 1, 2, ..., N) denotes the measurement data from the i^{th} training unit. The training dataset is used to train a predictive model. A test data set $\mathbf{y}_t$ denotes the measurement data from an online testing unit. The testing dataset is used to validate the predictive model. A weight vector $\mathbf{w}^{s_t} = [w_1^{s_t}, w_2^{s_t}, \ldots, w_M^{s_t}]^T$ denotes the weights associated with the degradation stage s_t ($s_t \in \{1, 2, \ldots, S\}$) of the testing unit, where M denotes the number of member algorithms. The ensemble prognostics approach first classifies the online testing unit, based on a sensed sample of the observable responses (or test dataset $\mathbf{y}_t$), into one of the predefined degradation stages, s_t, and then aggregates the prognostics results with degradation-stage associated weights $\mathbf{w}^{s_t}$.

Formulation of Ensemble Prognostics with Degradation-Dependent Weights

The predicted RULs of an online testing unit $\mathbf{y}_t$ by M member algorithms were aggregated to generate the ensemble-predicted RUL for the testing unit using the following weighted-sum formulation (Hu et al. 2012b):

$$\hat{L} = \sum_{j=1}^{M} w_j^{s_t} \hat{L}_j (\mathbf{y}_t, \mathbf{Y}) \tag{6}$$

where $\mathbf{y}_t$ denotes the measurement data from the online testing unit, $\hat{L}$ denotes the ensemble-predicted RUL for $\mathbf{y}_t$, $w_j^{s_t}$ denotes the weight assigned to the j^{th} prognostic algorithm associated with the degradation stage s_t, and $\hat{L}_j(\mathbf{y}_t, \mathbf{Y})$ denotes the predicted RUL by the j^{th} prognostic member algorithm trained with the data set $\mathbf{Y}$. Let the weight vector $\mathbf{w}^{st} = [w_1^{s_t}, w_2^{s_t}, \ldots, w_M^{s_t}]^{\mathrm{T}}$ and the vector of predicted RULs by member algorithms $\hat{\mathbf{L}} = \left[\hat{L}_1, \hat{L}_2, \ldots, \hat{L}_M\right]^{\mathrm{T}}$, the weighted-sum formulation in equation (6) can be expressed in a vector form as $\hat{L}\left(\mathbf{w}^{s_t}, \hat{\mathbf{L}}\right) = (\mathbf{w}^{s_t})^{\mathrm{T}} \hat{\mathbf{L}}$.

Optimization of Degradation-Dependent Weights

Achieving highly accurate and robust failure prognostics with the ensemble approach required assigning optimum weight values to the member algorithms for each of the S degradation stages. In the previous research (Hu et al. 2012b), an optimization-based weighting scheme was proposed to maximize the accuracy and robustness of an ensemble by synthesizing the prediction accuracy and diversity of its member algorithms. In this study, the optimization-based weighting scheme was used to optimize the degradation-dependent weight vectors. The weights for the s^{th} degradation stage can be determined by solving the following optimization problem:

$$\begin{aligned} \underset{\mathbf{w}^s}{\text{Minimize}} \quad & \varepsilon = S\left(\hat{L}_i^s\left(\mathbf{w}^s, \hat{\mathbf{L}}^s\left(\mathbf{y}_i\right)\right), L_i^T,\ i \in \mathbf{I}^s\right) \\ \text{Subject to} \quad & \sum_{j=1}^{M} w_j^s = 1 \end{aligned} \tag{7}$$

where $S(\bullet)$ is a predefined evaluation metric that measures the accuracy of the ensemble-predicted RUL, L_i^T denotes the true RUL of the i^{th} system unit, and $\mathbf{I}^s$ is an index set that contains the indices of all training units whose degradation stages are s.

Expectations were that the resulting ensemble with optimized degradation-dependent weights would optimally combine the generalization capabilities of the member algorithms and achieve robust RUL prediction. Given a pool of a large number of prognostic algorithms, solving the weight optimization problem in equation (7) would allow an optimal selection of prognostic algorithms from the pool to be used in the online testing phase. This is because larger weights associated with a degradation stage are likely to be assigned to algorithms that produce higher prognostic accuracy and diversity in the stage, while near-zero weights associated with the

degradation stages are likely to be assigned to algorithms that perform significantly more poorly than other algorithms in the stage.

CASE STUDY ON CAPACITY ESTIMATION

The efficacy of the deep learning approach developed under Module 1 was verified based on experimental data from a 10-year-long continuous cycling test (i.e., repeated full charge/discharge cycles) on eight Li-ion prismatic cells (Hu et al. 2018) used in an implantable application. This section first introduces the extraction of partial charge data from the full charge data, then discusses the implementation of training and validation and test for DCNNs, and finally presents and discusses the capacity estimation results.

Data Generation for Capacity Estimation

It was noted that in practice a Li-ion battery cell often does not experience a complete discharge process before being installed on a battery charger. A user often wishes to charge the cell before it has been completely depleted. In this case, the cell starts to be charged from a partially discharged state with a certain amount of remaining capacity. In order to simulate this real life case where the cell starts to be charged at the partially discharged state and ends up being fully charged, the researchers generated a partial charge curve from a full charge curve by truncating the full charge curve below a pre-assigned initial charge voltage (V_{initial}) (Hu et al. 2015, Hu et al. 2018). The value of V_{initial} was randomly drawn from a uniform distribution between a lower bound V_{low} and an upper bound V_{high}. This study considered two settings with different voltage ranges that corresponded to a low initial SOC and a high initial SOC to investigate how the initial SOC affected the accuracy of capacity estimation. Note that the high initial SOC setting indicated a cell undergoes a shallower discharge, as compared to the low initial SOC setting. Table 1 summarizes the voltage and SOC ranges for these two initial SOC settings.

Table 1. The two initial SOC settings considered in generation of partial charge data

Setting	Setting name	Voltage range	SOC range
Setting I	Low initial SOC	V_{low} = 3.65 V V_{high} = 3.80 V	Roughly 3%–23%
Setting II	High initial SOC	V_{low} = 3.80 V V_{high} = 3.85 V	Roughly 23%–43%

In addition to the effect of the initial SOC, the effect of current measurement error was also investigated in this study. This investigation considered two scenarios: Scenario I — No bias in current measurement and Scenario II — 2% positive bias in current measurement. In Scenario I, no change was made to the current and charge capacity measurements, whereas in Scenario II, the current-related features, the current and charge capacity, were artificially increased by 2% to simulate a 2% positive bias in the current measurement. The two scenarios were employed to study the influence of current measurement error on the accuracy of capacity estimation. Based on these two scenarios and the two voltage settings, a total of 4 different cases were considered in this study and are listed as follows:

- Case 1 — Low initial SOC (3%–23%), no bias in current measurement
- Case 2 — Low initial SOC (3%–23%), 2% positive bias in current measurement
- Case 3 — High initial SOC (23%–43%), no bias in current measurement
- Case 4 — High initial SOC (23%–43%), 2% positive bias in current measurement

Implementation of Training for DCNNs

The objective of the proposed deep learning method was to reduce the expected generalization error given by equation (2). In order to meet this objective, the researchers trained this DCNNs model using SGDM with a minibatch of 128 examples and their associated targets. An initial learning rate of 0.01 was set for all layers and this rate decreased by a factor of 5 for every 7 training epochs. The values of several important parameters used in training the DCNNs model appear in Table 2.

Table 2. List of parameter values used in training

Parameter	Value
Initial learning rate, α	0.01
Minibatch size	128
Momentum, γ	0.9
L_2 Regularization, λ	0.0001
Number of epochs	35

During the training process, the training and validation datasets were shuffled for each epoch to achieve a relatively unbiased estimator of the true gradient of the cost function and to give all samples an equal chance to be used. Specifically, the minibatch with 128 samples in one epoch, used for updating the parameters $\boldsymbol{\theta}$ (weights and biases) and computing the validation root mean square error (RMSE), is different from that in another epoch. For several reasons, including the randomness in the data shuffling and parameter initialization, two runs with the same training and validation datasets may give rise to two different local minima in the cost function. However, this is rarely a real concern and the usual course is to settle for finding a point in the parameter space that has a low cost but not the minimal cost (Saxe et al. 2013, Dauphin et al. 2014, Dauphin et al. 2015).

In this study, computations were carried out on a processor Intel Core i7-8700 CPU 3.2 GHz and 64 RAM, and an NVIDIA TITAN XP graphics processing unit (GPU) with 12 GB of GDDR5X memory. With the support from the advanced CPU and GPU, the network took between 5 and 7 minutes to train.

Implementation of Validation and Test for DCNNs

The conventional data splitting strategy is to first shuffle the entire dataset and then split it into training, validation, and test subsets to ensure that samples are selected randomly for training,

validation, and test. This is because many datasets are naturally arranged in some forms where successive samples are more likely correlated with their neighbors than those arranged farther away. The dataset in this study consisted of cycling test data from eight cells, arranged in descending order of the charge/discharge cycle number within a cell, and in ascending order of the cell number (i.e., cell 1 to cell 8). If samples in a minibatch were drawn following the order of this dataset, the minibatch would likely consist of samples from one specific cell and thus be biased toward the cell. In such a case, while the order of the dataset holds some physical significance, it is necessary to shuffle the dataset before splitting it for training and validation (Dauphin et al. 2015).

For the task of online capacity estimation, it was of practical significance for a trained DCNNs model to be capable of accurately estimating the capacity of one specific cell from the beginning to the end of life. In other words, the test dataset should contain complete data from one or multiple cells rather than partial data from the cell(s) used for the test. Consequently, the traditional splitting strategy needed to be adjusted for the specific task considered in this study. To this end, instead of shuffling the entire dataset, the researchers first took the data from one cell out of the entire dataset, to be used as the test set, and then shuffled the remaining data from the other seven cells to create the training (70%) and validation (30%) sets. The performance of the proposed method was evaluated using a test dataset that consisted of the complete data from one cell and was separated from the data used for training and validation.

Although the data splitting strategy adopted in this study differed from those usually used by researchers in the deep learning community, it allowed for evaluation of the performance of a trained deep learning model against a complete set of measurement data from the beginning to the end of life for any battery cell.

Definition of Error Measures

The accuracy of a trained DCNNs model was evaluated by using the eight-fold cross validation, as shown in Figure 5.

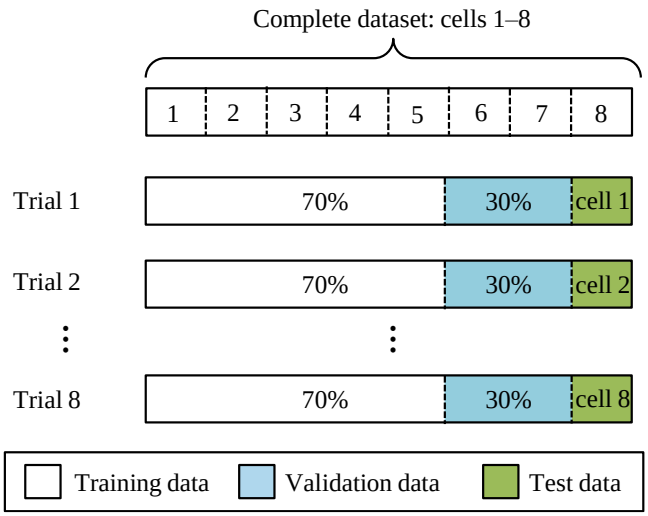

Figure 5. Procedure of eight-fold cross validation

The completed dataset was first split into eight mutually exclusive subsets or folds, cell 1, cell 2, …, cell 8, corresponding to the eight experimental cells. Subsequently, eight trials of training and validation and test were performed such that within each trial a fold of the data was held out for test, while the remaining seven folds were pulled together, randomly shuffled, and divided into a training dataset (70% of all samples in the seven folds) and a validation dataset (30%).

In the k^{th} cross validation trial, the trained DCNNs model was used to estimate the capacities of the samples in the test dataset (i.e., the complete data from the k^{th} cell). After performing all the eight cross validation trials, the overall test error $\varepsilon_{\text{RMS}}^{\text{All}}$ by DCNNs was then estimated by taking the average of the individual test errors $\varepsilon_{\text{RMS}}^{\text{k}}$ across the eight trials.

$$\varepsilon_{\text{RMS}}^{k} = \sqrt{\frac{1}{N_k}\sum_{i=1}^{N_k}(y^k(\mathbf{x}_i^k) - \hat{y}^k(\mathbf{x}_i^k))^2} \tag{8}$$

$$\varepsilon_{\text{RMS}}^{\text{All}} = \sqrt{\frac{1}{\sum_{k=1}^{8} N_k}\sum_{k=1}^{8}\sum_{i=1}^{N_k}(y^k(\mathbf{x}_i^k) - \hat{y}^k(\mathbf{x}_i^k))^2} \tag{9}$$

where N_k indicates the number of samples used for the test in the k^{th} cross validation trial, $\mathbf{x}_i^k$ is the input feature vector of the i^{th} sample in the k^{th} trial, and $y^k(\mathbf{x}_i^k)$ and $\hat{y}^k(\mathbf{x}_i^k)$ are the measured (or true) and estimated capacities for the i^{th} sample in the k^{th} trial, respectively.

Capacity Estimation Results

In order to estimate the accuracy of the proposed methodology for capacity estimation, a traditional machine learning method, relevance vector machine (RVM) (Hu et al. 2015, Hu et al. 2018), was also employed for the four cases explained earlier. Tables 3 through 6 show the

comparison results between RVM and DCNNs in terms of both the RMSE and maximum error (ME).

Table 3. Capacity estimation results by RVM and DCNNs for Case 1

Model	Item	#1	#2	#3	#4	#5	#6	#7	#8	Overall
RVM	RMSE	0.3379	0.3545	0.5230	0.4419	0.4691	0.4025	0.4203	0.3988	0.4185
	ME	4.5305	2.1429	3.0123	2.6030	2.9161	2.4677	2.4883	2.8807	4.5305
DCNNs	RMSE	0.3022	0.2980	0.3089	0.3677	0.4098	0.3117	0.2770	0.2619	0.3171
	ME	2.4910	2.6828	3.3294	3.5241	2.2963	2.5342	2.8852	2.0273	3.5241

Table 4. Capacity estimation results by RVM and DCNNs for Case 2

Model	Item	#1	#2	#3	#4	#5	#6	#7	#8	Overall
RVM	RMSE	0.3606	0.4411	0.3843	0.5339	0.5062	0.3636	0.3816	0.3706	0.4177
	ME	4.0473	3.2248	2.9912	3.0050	3.2329	3.4885	2.6736	2.9266	4.0473
DCNNs	RMSE	0.3583	0.2599	0.3217	0.4451	0.3923	0.3374	0.2787	0.2714	0.3331
	ME	2.8154	2.3472	4.4404	4.9244	2.1295	2.8925	4.2610	2.0590	4.9244

Table 5. Capacity estimation results by RVM and DCNNs for Case 3

Model	Item	#1	#2	#3	#4	#5	#6	#7	#8	Overall
RVM	RMSE	0.4734	0.4924	0.5742	0.5582	0.8408	0.5601	0.7613	0.6037	0.6080
	ME	5.4080	2.0123	6.2421	6.2263	5.6261	7.6505	3.6241	2.2397	7.6505
DCNNs	RMSE	0.3396	0.3604	0.3384	0.4700	0.3922	0.2776	0.3228	0.3076	0.3511
	ME	4.3461	2.8369	6.2855	6.4279	4.5568	5.8600	2.6983	2.1923	6.4279

Table 6. Capacity estimation results by RVM and DCNNs for Case 4

Model	Item	#1	#2	#3	#4	#5	#6	#7	#8	Overall
RVM	RMSE	0.4189	0.5540	0.5691	0.6474	0.8813	0.6394	0.6619	0.6007	0.6216
	ME	5.0855	2.1475	6.1685	6.0097	3.5672	8.6117	3.3122	2.5260	8.6117
DCNNs	RMSE	0.3408	0.3710	0.3937	0.4204	0.4919	0.2876	0.2704	0.4252	0.3752
	ME	4.6058	2.7902	5.9427	6.5492	4.7832	4.1072	3.7108	2.4969	6.5492

It should be noted that the capacity estimation of DCNNs was based on the best hypotheses of the proposed model (i.e., the best model with the lowest validation error among 150 runs).

Three significant observations were made from these results and are listed as follows:

- First, based on the overall RMSEs and Max errors from all eight cells, the performance of DCNNs was evidently better than that of RVM regardless of the SOC and current bias conditions. From the perspective of RMSEs and Max errors of individual cells, the results

suggested that DCNNs were capable of making a more accurate capacity estimation than RVM for all test cells. Although there were several situations where the Max error of RVM was slightly lower than the corresponding Max error of DCNNs, the overall performance of DCNNs was clearly better than RVM in all four cases. Furthermore, the team observed that both DCNNs and RVM were able to adapt the regression model to changes in the input vectors for all four cases, and the performance improvement of DCNNs over RVM was more significant in the high SOC than the low SOC.

- Second, introducing the 2% positive bias in current measurement did not significantly influence the accuracy of DCNNs (compare Case 1 versus Case 2 and Case 3 versus Case 4).

- Third, the RMSE by the proposed method did not show large increases when the initial SOC range was elevated (10.7% increase from Case 1 to Case 3 and 12.6% increase from Case 2 to Case 4). However, much larger increases were observed in the RMSE by RVM (45.3% increase from Case 1 to Case 3 and 48.8% increase from Case 2 to Case 4). Therefore, it can be concluded that neither deep (low initial SOC 3%–23%) nor shallow (high initial SOC 23%–43%) discharge conditions affected the capacity of the DCNNs to make an accurate capacity estimation.

In summary, the verification results based on 10 years of cycling data suggested that the DCNNs model was able to achieve a more accurate capacity estimation than RVM regardless of the initial charge level and the current measurement noise level. Indeed, DCNNs were more effective than RVM, because they possessed a higher generalization ability and were able to leverage a larger body of information.

Effect of Number of Layers

A convolutional neural network can consist of one or multiple convolutional layers. The optimal number of convolutional layers depends on the amount and complexity of the data. A parametric study was implemented to empirically investigate the effect of the number of convolutional layers on the accuracy of the DCNNs. For each cross validation trial of Cases 1 and 3, the team conducted 150 independent optimization runs with the number of convolutional layers varied from 0 to 6. Two box plots of the overall RMSE versus the number of convolutional layers were produced to graphically summarize the simulation results and are shown in Figure 6.

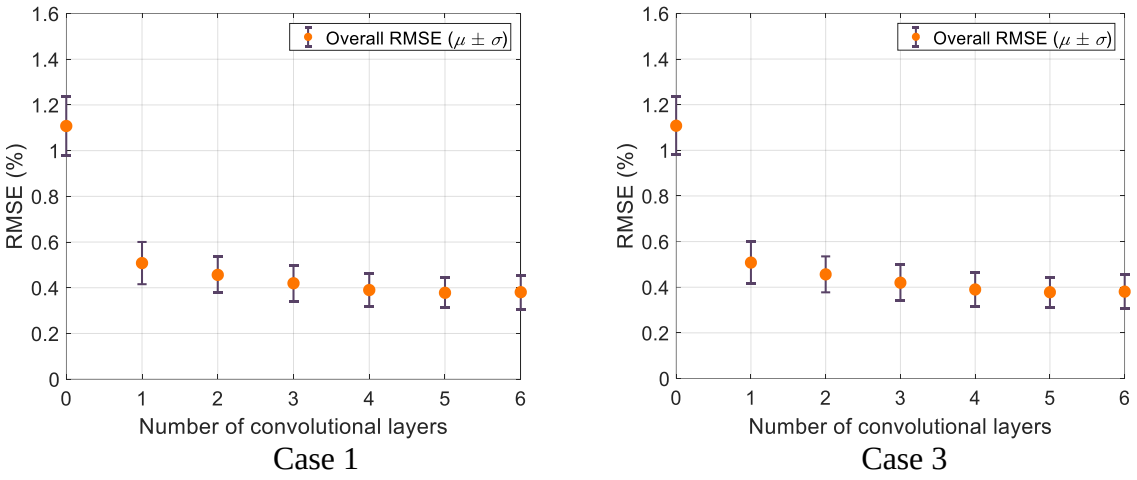

Figure 6. Overall RMSE on different numbers of convolutional layers for Case 1 and Case 3

Note that the overall RMSEs were based on the best 30 results out of the 150 runs. In DCNNs, a large number of convolutional layers may lead to a very small size of outputs at the last layer. In order to prevent the output size of the last convolutional layer from becoming too small, the maximum number of convolutional layers attempted in this study was 6.

Three observations can be made from the plots:

- Deep neural networks (i.e., DCNNs with 4 or more convolutional layers) significantly improve the accuracy in capacity estimation over the conventional neural networks (0 convolutional layers)

- Although each convolutional layer in a DCNNs model contains less than 1 percent of the networks' parameters, the number of convolutional layers plays a significant role in the estimation accuracy

- An inappropriate selection of the number of convolutional layers may lead to low estimation accuracy

The DCNNs model with five convolutional layers produced a slightly lower error when compared with the model with any other number of layers. Thus, the final DCNNs model was chosen to contain five convolutional layers and this network depth seemed to ensure satisfactory performance of the proposed method. The earlier results dealt (Tables 3 through 6) were derived using this chosen value of 5.

PROJECT RESULTS AND ACCOMPLISHMENTS

Results and Conclusions

An intelligent prognostics platform was developed for on-board SOH estimation and RUL prediction of Li-ion batteries. The uniqueness of the platform lies in its capabilities to automatically learn features of fade from large volumes of voltage and current measurement data during partial charge cycles (Module 1) and to synthesize the generalization strengths of multiple prognostic algorithms to ensure high prediction accuracy for an expanded range of battery applications and their operating conditions (Module 2).

To the knowledge of the principal investigator (PI), the research conducted under Module 1 represented the first attempt to apply deep learning to on-board SOH assessment of Li-ion batteries. The performance of the proposed deep learning approach was verified using 10 years of continuous cycling data acquired from implantable-grade Li-ion battery cells. The verification demonstrated that the proposed approaches achieved satisfactory accuracy in capacity estimation and outperformed a traditional machine learning-based approach, suggesting that the approach is a promising tool for online health management of Li-ion batteries.

Opportunity for Training and Development

The opportunities for training and development resulting from this project were as follows:

- The PI incorporated the research findings on failure prognostics as part of regular lectures within a new Iowa State University graduate-level course, ME 591X: Probabilistic Engineering Analysis and Design.

- The PI has shared the results of this research at invited seminars at Medtronic and Iowa State University.

- Three undergraduate students, Ha Lim Jeong (senior in mechanical engineering [ME]), Cole Tenold (senior in ME), and Stetsen Greiner (first year honors in ME), used this research project to gain research experience on the reliability and safety of Li-ion batteries. This project also enabled the PI and collaborators to provide research training for three graduate students: Yifei Li (MS in ME), Charlie Hubbard (MS in electrical and computer engineering), and Sheng Shen (PhD in ME).

Dissemination of Results

The results from this project have been disseminated through oral presentations by the PI and the students at three international conferences: the American Society of Mechanical Engineers (ASME) Design Automation Conference, Innovations in Biomedical Materials sponsored by the American Ceramic Society, and the Annual Conference of the Prognostics and Health

Management (PHM) Society. The PI helped organize the former two conferences by serving as symposium co-chair/track chair.

The researchers have made efforts to disseminate research results through collaborations with industry. Specifically, the researchers have been closely working with a leading medical device company, Medtronic, in the area of battery prognostics. Holding seminars and seeking internship positions for graduate students are two options to further this collaboration.

Products

The journal/conference papers and conference presentations that were partially supported by this project are listed as follows:

Hu, C., M. Hong, Y. Li, and H. Jeong. 2016. On-Board Analysis of Degradation Mechanisms of Lithium-Ion Battery using Differential Voltage Analysis. Paper presented at the Innovations in Biomedical Materials Conference, July 29–31, Chicago, IL.

——. 2016. On-Board Analysis of Degradation Mechanisms of Lithium-Ion Battery using Differential Voltage Analysis. Paper presented at the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, August 21–24, Charlotte, NC.

Hubbard, C., J. Bavslik, C. Hegde, and C. Hu. 2016. Data-Driven Prognostics of Li-Ion Rechargeable Battery using Bilinear Kernel Regression. Paper presented at the Annual Conference of the Prognostics and Health Management (PHM) Society, October 3–6, Denver, CO.

Li, Z., D. Wu, C. Hu, and J. Terpenny. 2019. An Ensemble Learning-Based Prognostic Approach with Degradation-Dependent Weights for Remaining Useful Life Prediction. *Reliability Engineering and System Safety*, Vol. 184, pp. 110–122. https://www.sciencedirect.com/science/article/pii/S0951832017308104.

——. 2017. An Ensemble Learning-based Prognostic Approach with Degradation-Dependent Weights for Remaining Useful Life Prediction. *Reliability Engineering and System Safety*, In Press.

Li, Y., M. K. Sadoughi, Z. Li., and C. Hu. 2017. An Ensemble Bias-Correction Method with Adaptive Weights for Dynamic Modeling of Lithium-Ion Batteries. Paper presented at the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, August 6–9, Cleveland, OH.

Li, Z., D. Wu, C. Hu, J. Terpenny, and S. Shen. 2017. An Ensemble Learning-based Prognostic Approach with Degradation-Dependent Weights for Remaining Useful Life Prediction. Paper presented at the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, August 6–9, Cleveland, OH.

Shen, S., M. K. Sadoughi, X. Chen, M. Hong, and C. Hu. 2018. Online Estimation of Lithium-Ion Battery Capacity Using Deep Convolutional Neural Networks. *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE)*, Paper Number: DETC2018-86347, August 26–29 2018, Quebec City, Quebec, Canada. http://proceedings.asmedigitalcollection.asme.org/proceeding.aspx?articleid=2713204.

Study Impacts

This research pioneers a novel intelligent prognostics platform that has the potential to enable the predictive control/maintenance of Li-ion batteries in transportation applications. Employing a novel deep learning approach will provide a new way for BMS designers to build predictive models for on-board capacity estimation that are more accurate and easier to design than existing machine learning models. The approaches and tools in the intelligent prognostics platform will provide greater transparency into the current and future health of an operating battery cell, more cost-effective maintenance/control (M/C) strategies and improved safety, and opportunities for life extensions. The platform, if successfully implemented in BMS, will potentially lead to the development of a cost-effective, and highly reliable and safe energy storage solution to transportation electrification.

REFERENCES

Bai, G., P. Wang, C. Hu, and M. Pecht. 2014. A Generic Model-Free Approach for Lithium-Ion Battery Health Management. *Applied Energy*, Vol. 135, pp. 247–260.

Chen, C., R. Xiong, and W. Shen. 2018. A lithium-ion battery-in-the-loop approach to test and validate multiscale dual H infinity filters for state-of-charge and capacity estimation. *IEEE Transactions on Power Electronics*, Vol. 33, No. 1, pp. 332–342.

Coble, J. B. and J. W. Hines. 2008. Prognostic algorithm categorization with PHM challenge application. Paper presented at the International Conference on Prognostics and Health Management, October 6–9, Denver, CO.

Dauphin, Y., H. de Vries, and Y. Bengio. 2015. Equilibrated adaptive learning rates for non-convex optimization. In Proceedings of the 28th International Conference on Neural Information Processing Systems, December 7–12, Montreal, Canada. pp. 1504–1512.

Dauphin, Y. N., R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio. 2014. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In Proceedings of the 27th International Conference on Neural Information Processing Systems, December 8–13, Montreal, Canada. pp. 2933–2941.

Eddahech A., O. Briat, N. Bertrand, J. Y. Deletage, and J. M. Vinassa. 2012. Behavior and state of health monitoring of Li-ion batteries using impedance spectroscopy and recurring neural networks. *International Journal of Electrical Power & Energy Systems*, Vol. 42, No. 1, pp. 487–494.

Gebraeel N., M. Lawley, R. Li, and J. K. Ryan. 2005. Residual-Life Distributions from Component Degradation Signals: A Bayesian Approach. *IIE Transactions*, Vol. 37, pp. 543–557.

Gebraeel N. and J. Pan. 2008. Prognostic degradation models for computing and updating residual life distributions in a time-varying environment. *IEEE Transactions on Reliability*, Vol. 57, No. 4, pp. 539–550.

Goebel, K., N. Eklund, and P. Bonanni. 2006. Fusing competing prediction algorithms for prognostics. Paper presented at the 2006 IEEE Aerospace Conference, March 4-11, Big Sky, MT.

He, W., N. Williard, M. Osterman, and M. Pecht. 2011. Prognostics of lithium-ion batteries based on Dempster–Shafer theory and the Bayesian Monte Carlo method. *Journal of Power Sources*, Vol. 196, No. 23, pp. 10314–10321.

Heimes, F.O. 2008. Recurrent neural networks for remaining useful life estimation. Paper presented at the International Conference on Prognostics and Health Management, October 6–9, Denver, CO.

Hu, C., H. Ye, G. Jain, and C. Schmidt. 2018. Remaining Useful Life Assessment of Lithium-Ion Batteries in Implantable Medical Devices. *Journal of Power Sources*, Vol. 375, pp. 118–130.

Hu, C., G. Jain, C. Schmidt, C. Strief, and M. Sullivan. 2015. Online estimation of lithium-ion battery capacity using sparse Bayesian learning. *Journal of Power Sources*, Vol. 289, pp. 105-113.

Hu, C., B. D. Youn, and J. Chung. 2012a. A multiscale framework with extended Kalman filter for lithium-ion battery SOC and capacity estimation. *Applied Energy*, Vol. 92, pp. 694–704.

Hu, C., B. D. Youn, P. Wang, and J. T. Yoon. 2012b. Ensemble of data-driven prognostic algorithms for robust prediction of remaining useful life. *Reliability Engineering & System Safety*, Vol. 103, pp. 120–135.

Ioffe, S. and C. Szegedy, C. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning, July 6–11, Lille, France. pp. 448–456.

Kim, J., S. Lee, and B. H. Cho. 2012. Complementary cooperation algorithm based on DEKF combined with pattern recognition for SOC/capacity estimation and SOH prediction. *IEEE Transactions on Power Electronics*, Vol. 27, No. 1, pp. 436–451.

Lee, S., J. Kim, J. Lee, and B. H. Cho. 2008. State-of-charge and capacity estimation of lithium-ion battery using a new open-circuit voltage vs. state-of-charge. *Journal of Power Sources,* Vol. 185, No. 2, pp. 1367–1373.

Liao, L. and F. Köttig. 2014. Review of hybrid prognostics approaches for remaining useful life prediction of engineered systems, and an application to battery life prediction. *IEEE Transaction on Reliability*, Vol. 63, No. 1, p.p. 191–207.

——. 2016. A hybrid framework combining data-driven and model-based methods for system remaining useful life prediction. *Applied Soft Computing*, Vol. 44, pp. 191–199.

Lu, L., X. Han, J. Li, J. Hua, and M. Ouyang. 2013. A review on the key issues for lithium ion battery management in electric vehicles. *Journal of Power Sources*, Vol. 226, pp. 272–288.

Liu, J., A. Saxena, K. Goebel, B. Saha, and W. Wang. 2010. An Adaptive Recurrent Neural Network for Remaining Useful Life Prediction of Lithium-Ion Batteries. Paper presented at the Annual Conference of the Prognostics and Health Management Society, October 10–14, Portland, OR.

Liu, D., Y. Song, L. Li, H. Liao, and Y. Peng. 2018. On-line life cycle health assessment for lithium-ion battery in electric vehicles. *Journal of Cleaner Production*, Vol. 199, pp. 1050–1065.

Luo, J., K. R. Pattipati, L. Qiao, and S. Chigusa. 2008. Model-based prognostic techniques applied to a suspension system. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, Vol. 38, No. 5, pp. 1156–1168.

Miao, Q., L. Xie, H. Cui, W. Liang, and M. Pecht. 2013. Remaining useful life prediction of lithium-ion battery with unscented particle filter technique. *Microelectronics Reliability*, Vol. 53, No. 6, pp. 805–810.

Nair, V. and G. E. Hinton. 2010. Rectified linear units improve restricted Boltzmann machines. In Proceedings of the 27th International Conference on Machine Learning, June 21–24, Haifa, Israel. pp. 807–814.

Ng, K. S., C. S. Moo, Y. P. Chen, and Y. C. Hsieh. 2009. Enhanced coulomb counting method for estimating state-of-charge and state-of-health of lithium-ion batteries. *Applied Energy*, Vol. 86, No. 9, pp. 1506–1511.

Ng, S. S. Y., Y. Xing, K. L. Tsui. 2014. A naive Bayes model for robust remaining useful life prediction of lithium-ion battery. *Applied Energy*, Vol. 118, pp. 114–123.

Nuhic, A., T. Terzimehic, T. Soczka-Guth, M. Buchholz, and K. Dietmayer. 2013. Health diagnosis and remaining useful life prognostics of lithium-ion batteries using data-driven methods. *Journal of Power Sources*, Vol. 239, pp. 680–688.

Plett, G. L. 2004. Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs. *Journal of Power Sources*, Vol. 134, pp. 277–92.

——. 2006. Sigma-point Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 2: Simultaneous state and parameter estimation. *Journal of Power Sources*, Vol. 161, No. 2, pp. 1369–1384.

Polyak, B. T. 1964. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, Vol. 4, No. 5, pp.1–17.

Saha, B. and K. Goebel. 2009. Modeling Li-ion battery capacity depletion in a particle filtering framework. Paper presented at the Annual Conference of the Prognostic and Health Management Society, September 27–October 1, San Diego, CA.

Saha, B., K. Goebel, S. Poll, and J. Christophersen. 2009. Prognostics methods for battery health monitoring using a Bayesian framework. *IEEE Transactions on Instrumentation and Measurement*, Vol. 58, No. 2, pp. 291–296.

Saxe, A. M., J. L. McClelland, and S. Ganguli. 2013. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. preprint arXiv:1312.6120v3.

Schmidt, A. P., M. Bitzer, A. W. Imre, and L. Guzzella. 2010. Model-based distinction and quantification of capacity loss and rate capability fade in Li-ion batteries. *Journal of Power Sources*, Vol. 195, No. 22, pp. 7634–7638.

Shen, S., M. K. Sadoughi, X. Chen, M. Hong, and C. Hu. 2018. Online Estimation of Lithium-Ion Battery Capacity Using Deep Convolutional Neural Networks. *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE)*, Paper Number: DETC2018-86347, August 26–29 2018, Quebec City, Quebec, Canada. http://proceedings.asmedigitalcollection.asme.org/proceeding.aspx?articleid=2713204.

Waag, W. and D. U. Sauer. 2013. Adaptive estimation of the electromotive force of the lithium-ion battery after current interruption for an accurate state-of-charge and capacity determination. *Applied Energy*, Vol. 111, pp. 416–427.

Wang, D., Q. Miao, and M. Pecht. 2013. Prognostics of lithium-ion batteries based on relevance vectors and a conditional three-parameter capacity degradation model. *Journal of Power Sources*, Vol. 239, pp. 253–264.

Wang, D., F. Yang, K. L. Tsui, Q. Zhou, and S. J. Bae. 2016. Remaining useful life prediction of lithium-ion batteries based on spherical cubature particle filter. *IEEE Transactions on Instrumentation and Measurement*, Vol. 65, No. 6, pp. 1282–1291.

Wang, T., J. Yu, D. Siegel, and J. Lee. 2008. A similarity-based prognostics approach for remaining useful life estimation of engineered systems. International Conference on Prognostics and Health Management, October 6–9, Denver, CO.

Wang, X., N. Balakrishnan, and B. Guo. 2014. Residual life estimation based on a generalized Wiener degradation process. *Reliability Engineering & System Safety*, Vol. 124, pp. 13–23.

Xiong, R., F. Sun, Z. Chen, and H. He. 2014. A data-driven multi-scale extended Kalman filtering based parameter and state estimation approach of lithium-ion polymer battery in electric vehicles. *Applied Energy*, Vol. 113, pp. 463–476.

Zio, E. and F. Di Maio. 2010. A data-driven fuzzy approach for predicting the remaining useful life in dynamic failure scenarios of a nuclear system. *Reliability Engineering & System Safety*, Vol. 95, No. 1, pp. 49–57.

THE INSTITUTE FOR TRANSPORTATION IS THE FOCAL POINT FOR TRANSPORTATION AT IOWA STATE UNIVERSITY.

InTrans centers and programs perform transportation research and provide technology transfer services for government agencies and private companies;

InTrans manages its own education program for transportation students and provides K-12 resources; and

InTrans conducts local, regional, and national transportation services and continuing education programs.

IOWA STATE UNIVERSITY

Visit www.InTrans.iastate.edu for color pdfs of this and other research reports.