Remaining Useful Life Prediction of Lithium-ion Battery based on Attention Mechanism with Positional Encoding

Beitong Zhou¹, Cheng Cheng¹, Guijun Ma² and Yong Zhang³

¹ School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan, 430074, China
² School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan, 430074, China
³ School of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan, 430081, China

E-mail: zhoubt@hust.edu.cn

Abstract. The rising demands of more reliable and stable electrical systems attach importance to accurate Remaining Useful Life (RUL) prediction of the lithium-ion batteries. As artificial intelligence and machine learning techniques advance, data-driven methods especially deep learning algorithms have become the rising star in RUL prediction. Recurrent Neural Networks (RNNs) and their variants such as Long Short Term Memory have proven effectiveness in various sequential tasks. However, due to its iterative nature along the time axis, RNNs take much time for information to flow through the network for prediction. Inspired by recent advance brought by Transformer in sequence transduction tasks, we proposed the attention mechanism based Convolutional Neural Network (CNN) with positional encoding to tackle this problem. The attention mechanism enables the network to focus on specific parts of sequences and positional encoding injects position information while utilizing the parallelization merits of CNN on GPUs. Empirical experiments show that the proposed approach is both time effective and accurate in battery RUL prediction.

1. Introduction

Due to the high energy density, no memory effect, and low self-discharge traits of lithium-ion batteries, they are becoming widely used in a large amount of energy systems including consumer electronics, electrical vehicles, and aerospace systems [1] [2]. However, with many merits, the batteries can harm the public safety because of a flammable electrolyte and can lead to explosions and unexpected damages. To help fully utilize the advantages of the lithium-ion batteries and prevent them from causing catastrophic damage to human safety, it’s necessary to accurately monitor the battery’s states. There are several important indicators such as State of Charge (SOC), State of Health (SOH), and Remaining Useful Life (RUL). The lithium-ion battery degrades through repeated charge cycles. The battery capacity, which is defined as the amount of electrical charge a battery can holds in its full state, decreases along with the degradation process. Thus, RUL of a battery indicates the number of remaining charge-discharge cycles before the states degrade to a given failure threshold. For practical use, the risk of causing damage increases when the capacity is reduced to below 70%-80% of its initial value and the battery fails to provide further use [3] [4] [5]. The accurately predicted RUL is of great importance for maintenance in advance and alarming the potential risk before the battery reaches the failure threshold.
Comprehensive research has been conducted on the battery modeling and RUL prediction. Those algorithms can be roughly divided into model-based methods and data-driven methods. The model based methods include equivalent circuit models, electrochemical models, and empirical models. The equivalent circuit model [6] [7] is formed by an open-circuit voltage source connected with electric elements, such as resistors and capacitors. Because the equivalent circuit model is empirical analysis, it depends on impedance data and cannot provide highly accurate prediction results. On the other hand, the electrochemical models [8] [9] reveals the internal physics property of the battery through partial differential equations (PDEs) but it is hard to solve PDEs in constant time. For empirical models, the capacity-based exponential models combined with filter approaches such as Kalman Filter (KF), Particle Filter (PF) and their variants are common solutions. Saha et al. [10] used the PF algorithm to predict the battery RUL and the uncertainty representation with the empirical degradation model as the state transition equation. Chang et al. [11] proposed a hybrid approach which combines the Unscented Kalman Filter (UKF) method with a double exponential model for long-term capacity prediction. Miao et al. [12] obtained the prediction results of the lithium-ion battery RUL by using UKF algorithm also based on the degradation model. Although empirical models are preferred from practical perspective, there are still demands of a more precise RUL prediction.

Data-driven methods, such as Support Vector Regression (SVR), Relevance Vector Machine (RVM), and Box-Cox Transformation (BCT), make predictions in another different way which learn latent representations directly from history data. Dong et al. [13] used the Support Vector Regression - Particle Filter (SVR-PF) algorithm to provide RUL prediction as well as SOH monitoring. Zhang et al. [14] developed an RUL prediction method based on the BCT and monte carlo simulation. Recently, as artificial intelligence and deep learning methods prevail in various tasks such as computer vision [15], image caption [16] and neural machine translation [17], Recurrent Neural Network (RNN) and its variants have proven its effectiveness in processing with sequential tasks including RUL predictions. Liu et al. [18] proposed an Adaptive Recurrent Neural Network (ARNN) for system dynamic state forecasting and its effectiveness was demonstrated in RUL prediction. Zhang et al. [19] employed Long Short-Term Memory (LSTM) to learn the long-term dependencies among the degraded capacities of lithium-ion batteries. Convolutional Neural Network, another popular deep learning network in image classification [20], has already been employed in mechanical systems such as fault diagnosis [21] [22]. Cheng et al. [23] proposed an online data-driven framework to exploit the adoption of deep CNN in predicting the RUL of bearings. Ma et al. [24] used a hybrid neural network that combines the advantages of CNN and LSTM to make RUL predictions.

Deep learning methods, which extract features from unstructured data automatically and have abilities of fitting complex datasets, have already outperformed model-based methods and traditional machine learning methods in RUL prediction. However, due to the iterative nature of LSTM, it can take a lot of time for information to flow through the network sequentially with long sequences, making it theoretically impossible to exploit the merits of concurrent hardwares. CNN, on the other hand, would not be affected by the length of input sequences and is prone to parallelization as information of each kernel within the same layer is inferred in isolation. A network, which can be implemented in parallel and able to keep position information within the input data, is of great importance to practical use. Being released in late 2017, the Transformer architecture proposed in [17] has refreshed multiple SOTA records in neural machine translation tasks. The so-called Transformer network leverages fully connected networks with a cleverly-designed attention mechanism, which indicates the directions of possible solutions to our above problems.

Inspired by Transformer, this paper proposes to use CNN based on attention mechanism with positional encoding to predict the RUL of lithium-ion batteries. The implementation contains following stages. Firstly, the sliding window approach with a given sequence length will process the raw data and the CNN layer acts as a low dimensional embedding to extract features from inputs. Secondly, position information will be injected into embeddings through positional encoding. Meanwhile, attention weights are computed to help the network focus on specific parts of intermediate information. Thirdly, a fully connected layer is used to achieve the final RUL prediction, where the RUL can be calculated once the predicted capacities reach the failure threshold.
This work makes the following contributions: 1) High accuracy. Empirical experiments are conducted to prove its effectiveness of making precise RUL predictions compared with other methods. 2) Attention mechanism and positional encoding are adopted. Positional encoding injects position information which can make up for the weakness CNN has compared with RNN. Attention mechanism help focus on specific parts and improve the robustness and performance of the network. 3) Easy to implement in parallel. The proposed algorithm overcomes the weakness of RNN when the sequence becomes long and can fully exploit the parallelization ability of GPUs.

The paper’s outline is as follow. Section 2 introduces the proposed algorithm framework. In section 3, experiments are conducted and detail analysis is included. Section 4 summarizes the whole work and indicates future works.

2. Algorithms
To solve the above-stated problem, the schematic of proposed framework is shown in Figure 1 which involves three main parts. Related algorithms will be discussed in detail.

![Figure 1. RUL prediction framework, which is mainly comprised of three parts: (1) Convolutional layer; (2) Positional encoding and attention mechanism; (3) Regression layer.](image)

2.1. Convolutional neural network
CNN proposed by Lecun et al. in works [25] and [26] becomes the most well-known deep learning architecture since its superior performance in computer vision tasks such as image classification [20] and object detection [27]. A typical CNN architecture mainly contains local connections, shared weights, and local pooling [28].

In this work, only a 1-D CNN layer is used for simplicity, since the input data are time sequences. Convolutional layer usually contains multiple kernels and every kernel has a filter \( w \in \mathbb{R}^f \) and a bias \( b \in \mathbb{R} \). The output feature \( v_i \) is obtained through a convolution operation, which is between an input kernel of size \( f \) and the filter \( w \), and a non-linear activation function ReLU to avoid potential vanishing gradient problem. The whole transformation for a single kernel is described as:

\[
v_i = ReLU(w * u_j + b)
\]

where \( w_j \in \mathbb{R}^{1 \times f} \) represents the \( j \)-th sub-vector of the input data, \( v_i \in \mathbb{R} \) is the \( i \)-th value in the convolution outputs and ‘\(*\)’ denotes the convolution operation. The outputs of the convolution operation \( v = [v_1, v_2, ..., v_o] \) is also called as the feature map, where \( o = (l - f)/s + 1 \) is the number of output features, \( l \) is the length of input sequence length and \( s \) is the stride for convolution. Usually for a typical convolutional layer with \( k \) kernels, we would have \( k \) independent feature maps stacked together.
2.2. Positional Encoding and Attention Mechanism

Positional encoding, which was first introduced in [17], injects positional information about the relative or absolute position of the sequence. In this paper, the implementation is the same as the original work except that we chose 1000 not 10000 due to lower dimensions and shorter sequence length:

\[ PE_{\text{pos}, 2i} = \sin(\text{pos}/1000^{2i/d}) \]

\[ PE_{\text{pos}, 2i+1} = \cos(\text{pos}/1000^{2i/d}) \]

where the pos is the position, i is the index of input dimensions and PE is the abbreviation of positional encoding. We visualized the positional encoding in heatmap in Figure 2. In the figure, each row which is sequentially ordered corresponds to the positional encoding of a vector. Adding this kind of position-dependent information may help the model incorporate the order of each positions.

Attention mechanism was first proposed in the work [29] which integrates attention mechanism with an encoder-decoder architecture on neural machine translation tasks. Many deep learning models in time series tasks are still shallow due to the lack of interpretability and large-scale data while attention mechanism provides a compelling solution. Lai et al. [30] proposed LSTMNet and was one of the first to combine LSTM with attention to forecast multivariate time series. Shih et al. [31] proposed a novel attention mechanism for selection and use frequency domain information for time series forecasting.

The output of positional encoding is \( k \in \mathbb{R}^{k \times l} \) and position weights \( s \in \mathbb{R}^{k \times l} \) are computed through attention layer. A fully connected layer with weights \( w \in \mathbb{R}^{l \times l} \) and a bias \( b \in \mathbb{R}^l \) is used to implement the modified attention mechanism. The outputs are normalized using the softmax function:

\[ s_{ij} = \frac{\exp(e_{ij})}{\sum_{m=1}^{l} \exp(e_{im})} \]

where \( e_{ij} = wh_i + b \). The final outputs of the attention layer is the element-wise production between the intermediate representations \( h \) and the position weights \( s \).

![Figure 2. The heatmap visualization of positional encoding. There are 100 rows and each row is a vector of length 500.](image)

2.3. Regression Layer

Finally, the outputs of attention layer are fed into the regression layer which is simply a fully connected linear layer without any activation function. For a regression problem, the commonly used mean squared error (MSE) function is chosen as the error metric to evaluate the model performance:

\[ e = \frac{1}{N} \sum_{m=1}^{N} (y_m - \hat{y}_m)^2 \]

where \( y_m \) is the ground label and \( \hat{y}_m \) is the predicted value. The goal of optimizing the whole network is to minimize the mean squared error by optimizers such as Adagrad [32] and Adam [33].

The overall framework of the proposed algorithm can be summarized in Algorithm 1.
Algorithm 1 Proposed algorithm.

Require: The train set $D_{\text{train}}$
Initialize: The weight parameters of convolutional layer, attention layer and the final regression layer
1: while not reach the maximum iterations do
2: Get degradation data from raw inputs using sliding window approach.
3: Feed degradation data into the convolutional layer and get the low dimensional embeddings $v$.
4: Add position information using positional approach.
5: Feed intermediate representations $h$ into the attention layer to compute position weights and outputs are the element-wise production of representations $h$ and the position weights $s$.
6: Use the final regression layer to make RUL predictions and calculate the MSE loss
7: Compute the gradients with Adam optimizer and update the parameters.
8: end while

Table 1. Evaluation Metrics.

| Metrics     | Equations |
|-------------|-----------|
| Error Metric| $EM = |R - \hat{R}|$ |
| Accuracy Metric | $AM = \left( 1 - \frac{|R - \hat{R}|}{R} \right) \times 100\%$ |
| Steady Metric | $SM = \sqrt{\frac{1}{n-k} \sum_{i=k+1}^{n} (x_i - \hat{x}_i)^2}$ |

2.4. Evaluation Metrics
We chose three evaluation metrics listed in Table 1 to estimate the model performance. The Error Metric (EM), Accuracy Metric (AM) and Steady Metric (SM) are frequently used metrics for comparisons.

In the table, $R$ and $\hat{R}$ represents the actual RUL which is the number of charge-discharge cycle when SOH reaches the failure threshold and the estimated RUL. For SM, $x_i$ and $\hat{x}_i$ are the actual capacity and the predicted capacity in test set $D_{\text{test}}$. $k$ is the number of cycles in train set $D_{\text{train}}$ and the $n$ is the total number of cycles for the dataset. A model which can achieve accurate predictions is supposed to have small enough EM and SM while AM should be large.

3. Experiments

3.1. SOH and RUL
State of Health (SOH) is a commonly used indicator of the battery’s operating states in a repeated charge-discharge cycle. SOH can be calculated using capacity, internal resistance, cycle number, etc. For simplicity, the capacity ratio is used to describe SOH. The SOH $SOH_k$ in cycle $k$ is described as:

$$SOH_k = \frac{x_k}{x_0} \times 100\%$$ (5)

where $x_0$ is the initial capacity and $x_k$ is the battery capacity in the number $k$ charge-discharge cycle. The battery RUL can be defined as the number of remaining char-discharge cycles in a certain time based on historical degradation curve.
3.2. Dataset Description
The dataset used in following experiments has been collected from a battery prognostics test bed at the NASA Ames Prognostics Center of Excellence (PCoE). The commercially available Li-ion 18650 sized rechargeable batteries with a Graphite anode and Lithium Nickel Cobalt Manganese oxide (LiNiCoMnO2) cathode are used. The test bed contains a programmable 4-channel Direct Current (DC) electronic load, programmable 4-channel DC power supply, voltmeter, ammeter, etc.. The lithium-ion batteries were run through three different operational profiles (charge, discharge and Electrochemical Impedance Spectroscopy) at different temperatures. The battery charge was carried out in a Constant Current (CC) mode at 1.5 A till the battery voltage reached 4.2 V, the continued in a Constant Voltage (CV) mode until the charge current dropped to 20 mA. Discharges were carried out at different current load levels until the battery voltage fell to preset voltage thresholds. Repeated charge and discharge cycles resulted in accelerated aging of batteries.

3.3. Results
For each lithium-ion battery, we keep using the same hyper-parameters such as kernel size, stride size, and learning rate to guarantee the stability and robustness. The model architecture is shown in Table 2.

| Table 2. Model Architecture. |
|-------------------------------|
| Input Size | Output Size | Layer Setting |
| Input Layer | 1x1x20 | ... | ... |
| Convolutional Layer | 1x1x20 | 1x8x16 | k=8, f=5, s=1 |
| Positional Encoding | 1x8x16 | 1x8x16 | ... |
| Attention Layer | 1x8x16 | 1x8x16 | 16 neurons |
| Flatten | 1x8x16 | 1x128 | ... |
| Regression Layer | 1x128 | 1x1 | 1 neuron |

| Table 3. Comparisons of predicted results between the proposed network and other existing deep learning methods. |
|------------------|--------|--------|--------|--------|
|                  | CNN    | LSTM   | CNN + Attention |
|                  | Rb     | EM     | AM     | SM     | Rb     | EM     | AM     | SM     | Rb     | EM     | AM     | SM     |
| Battery-1        | 88     | 73     | 61.64% | 0.0796 | 46     | 27     | 63.01% | 0.0579 | 72     | 1      | 98.6%  | 0.0108 |
| Battery-2        | 60     | 46     | 67.39% | 0.0369 | 26     | 29     | 56.52% | 0.0615 | 36     | 10     | 78.26% | 0.0217 |
| Battery-3        | 119    | 46     | 67.39% | 0.0304 | 58     | 12     | 73.91% | 0.0116 | 51     | 5      | 89.13% | 0.0057 |

For input data, the sliding window size is chosen as 20, corresponds to the input length l of convolutional layer. The whole network is optimized using Adam optimizer. The learning rate and weight decay are set as 0.001 and 0.005.
Figure 3. Prediction results of 3 different lithium-ion batteries in the NASA dataset. The sliding window size here is set to 20 so that no predictions are made in first 20 cycles. The monitoring thresholds, failure thresholds and their corresponding capacities are annotated in the figure. Battery-3 didn’t reach the failure threshold so that the lowest capacity was chosen as the failure threshold and the cycle of it was the actual RUL.

To conduct experiments to evaluate model performance, a monitoring threshold of 0.8 SOH was chosen to split train set $D_{\text{train}}$ and test set $D_{\text{test}}$. The actual RUL was defined as the charge-discharge cycles needed to reach the failure threshold of 0.7 SOH. There are 3 lithium-ion batteries in the NASA dataset and the prediction results are shown in Figure 3. Then network fits the train set well and predicts the RUL accurately as we can see in the Figure 3. To understand what the attention mechanism is actually doing, we randomly choose a input sequence from the Battery-3 and visualize the position weights in heatmaps. The visualization is shown in Figure 4.

Figure 4. Heatmap visualization for position weights in the attention layer given a random input. The convolutional layer reduces the sequence length from 20 to 16 and the dimensions increase to 8. The lightness in a given position indicates the weights in the attention mechanism.

In the above heatmap, we can notice that the layer pays more attention as the sequence reaches the end and for different dimensions, usually different weights are given. It shows that the attention mechanism can actually capture the position order and focus more on areas which are closer to the prediction label.
3.4. Comparisons
We also compare the proposed algorithm with other CNN and LSTM. The experiment including evaluation metrics results for two methods are listed in the work [24]. The final results are listed in Table 3. As we can see, our method achieves the lowest EM, SM and the highest AM. It shows that compared with other methods, the proposed algorithm predicts the RUL more accurately and steadily. Another comparison experiment is conducted to verify the time effectiveness of CNN. A stacked LSTM with 2 layers and 8 hidden units is chosen as the comparing object. Randomized inputs with different input lengths 10,50,100,200,500 are fed into two networks and running for 100 times. The average inference time are recorded and the experiments are conducted using Intel(R) Core(TM) i74770HQ CPU in macOS Catalina. The results are shown in Table 4. The increased input length only adds marginal time cost to CNN inference time but for LSTM, the cost can be hard to ignore.

Table 4. Average inference time for two networks.

| Model            | Input Length |
|------------------|--------------|
| CNN + Attention  | 0.485ms 2.189ms 2.512ms 2.584ms 4.024ms |
| Stacked LSTM     | 2.007ms 45.15ms 85.93ms 175.3ms 409.7ms |

4. Conclusion
In this work, we propose a convolutional neural network based on attention mechanism with positional encoding to predict the RUL of lithium-ion batteries. The convolutional layer is used to extract features and translate the inputs into a low dimensional space. The positional encoding injects position information using a encoding function which is relevant to the dimension and sequence position. The attention mechanism helps focus on specific parts of intermediate representations. Finally, a regression layer is used to make RUL predictions. Results on the NASA dataset shows that the proposed network can indeed capture hidden trending and make accurate RUL predictions. We also compare the averaged inference time with LSTM to verify the superior advantage of CNN for practical use. Future work includes: 1) find out a more reliable way to cross validate the model performance and 2) improve the algorithm to reduce the instability brought by random optimization.

5. References
[1] Liu X, Wu J, Zhang C and Chen Z 2014 *Journal of Power Sources* 270 151–157
[2] Tarascon J M and Armand M 2011 Issues and challenges facing rechargeable lithium batteries *Materials for Sustainable Energy: A Collection of Peer-Reviewed Research and Review Articles from Nature Publishing Group* (World Scientific) pp 171–179
[3] Liu D, Wang H, Peng Y, Xie W and Liao H 2013 *Energies* 6 3654–3668
[4] Tang S, Yu C, Wang X, Guo X and Si X 2014 *Energies* 7 520–547
[5] Dubarry M and Liaw B Y 2009 *Journal of Power Sources* 194 541–549
[6] Hu X, Li S and Peng H 2012 *Journal of Power Sources* 198 359–367
[7] Remmlinger J, Buchholz M, Meiler M, Bernreuter P and Dietmayer K 2011 *Journal of Power Sources* 196 5357–5363
[8] Zhang W J 2011 *Journal of Power Sources* 196 13–24
[9] Kemper P, Li S E and Kum D 2015 *Journal of Power Sources* 286 510–525
[10] Saha B, Goebel K and Christophersen J 2009 *Transactions of the Institute of Measurement and Control* 31 293–308
[11] Chang Y, Fang H and Zhang Y 2017 *Applied energy* 206 1564–1578
[12] Miao Q, Xie L, Cui H, Liang W and Pecht M 2013 *Microelectronics Reliability* 53 805–810
[13] Dong H, Jin X, Lou Y and Wang C 2014 *Journal of power sources* 271 114–123
[14] Zhang Y, Xiong R, He H and Pecht M G 2018 IEEE Transactions on Industrial Electronics 66 1585–1597

[15] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition Proceedings of the IEEE conference on computer vision and pattern recognition pp 770–778

[16] Jia X, Gavves E, Fernando B and Tuytelaars T 2015 Guiding the long-short term memory model for image caption generation Proceedings of the IEEE international conference on computer vision pp 2407–2415

[17] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser Ł and Polosukhin I 2017 Attention is all you need Advances in neural information processing systems pp 5998–6008

[18] Liu J, Saxena A, Goebel K, Saha B and Wang W 2010 An adaptive recurrent neural network for remaining useful life prediction of lithium-ion batteries Tech. rep. NATIONAL AERONAUTICS AND SPACE ADMINISTRATION MOFFETT FIELD CA AMES RESEARCH ...

[19] Zhang Y, Xiong R, He H and Pecht M G 2018 IEEE Transactions on Vehicular Technology 67 5695–5705

[20] Krizhevsky A, Sutskever I and Hinton G E 2012 Imagenet classification with deep convolutional neural networks Advances in neural information processing systems pp 1097–1105

[21] Jing L, Zhao M, Li P and Xu X 2017 Measurement 111 1–10

[22] Wen L, Li X, Gao L and Zhang Y 2017 IEEE Transactions on Industrial Electronics 65 5990–5998

[23] Cheng C, Ma G, Zhang Y, Sun M, Teng F, Ding H and Yuan Y 2018 CoRR abs/1812.03315 (Preprint 1812.03315) URL http://arxiv.org/abs/1812.03315

[24] Ma G, Zhang Y, Cheng C, Zhou B, Hu P and Yuan Y 2019 Applied Energy 253 113626

[25] LeCun Y, Boser B, Denker J S, Henderson D, Howard R E, Hubbard W E and Jackel L D 1990 Handwritten digit recognition with a back-propagation network Advances in Neural Information Processing Systems pp 396–404

[26] LeCun Y, Boser B, Denker J S, Henderson D, Howard R E, Hubbard W and Jackel L D 1989 Neural Computation 1 541–551

[27] Girshick R, Donahue J, Darrell T and Malik J 2014 Rich feature hierarchies for accurate object detection and semantic segmentation Proceedings of the IEEE conference on computer vision and pattern recognition pp 580–587

[28] Saxe A M, Koh P W, Chen Z, Bhand M, Suresh B and Ng A Y 2011 On random weights and unsupervised feature learning Proceedings of the 28th International Conference on International Conference on Machine Learning (Omnipress) pp 1089–1096

[29] Bahdanau D, Cho K and Bengio Y 2014 arXiv preprint arXiv:1409.0473

[30] Lai G, Chang W C, Yang Y and Liu H 2018 Modeling long-and short-term temporal patterns with deep neural networks The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (ACM) pp 95–104

[31] Shih S Y, Sun F K and Lee H y 2019 Machine Learning 108 1421–1441

[32] Duchi J, Hazan E and Singer Y 2011 Journal of Machine Learning Research 12 2121–2159

[33] Kingma D P and Ba J 2014 arXiv preprint arXiv:1412.6980

Acknowledgments

Authors would like to thank Prof. Ye Yuan for his valuable comments and helpful suggestions. This work was supported by the Key Laboratory of Image Processing and Intelligent Control of China under Grant IPIC2019-08 and by the National Key R&D Program of China under Grant 2018YFB1701200.