Manuscript ID ER-21-19075

Paper Review Reponses

May 3rd, 2021

Dear reviewers and editors:

We would like to sincerely thank you for your valuable comments on our paper, manuscript ID ER-21-19075, entitled "Multivariate Gated Recurrent Unit for Battery Remaining Useful Life Prediction: A Deep Learning Approach". Your comments have been indispensable in our efforts to improve the quality of this paper, and make it more complete in terms of both the technical aspects and readability.

We have revised our paper, taking into full consideration the reviewers' and editor's comments and suggestions. Below are our review responses (16 pages in total including this page). All of the revisions that respond to the questions or comments from the reviews are highlighted in yellow in the newly uploaded manuscript, while the original content of the previous version that relates to the review comments is highlighted in green.

Yours very truly,
Authors of the paper

Encl.:

Reviewer: 1

Comments to the Author

The work is somehow new but several limitations from the English to the technical hinder to grasp the main point of the work, some of them are listed below:

Response: Many thanks for your valuable comment.

1. In my opinion, the abstract is too cumbersome, and is hard to catch the key point. The keywords need to be more detailed.

Response: Thanks for your insightful comment. We have rewritten the abstract.

Summary

This paper proposes the gated recurrent unit (GRU)-recurrent neural network (RNN), a deep learning approach to predict the remaining useful life (RUL) of Lithium-ion batteries, accurately. The GRU-RNN structure can self-learn the network parameters by utilizing adaptive gradient descent algorithms, leading to a reduced computational cost. Unlike the long short-term memory (LSTM) model, GRU-RNN allows time-series dependencies to be tracked between degraded capacities without using any memory cell. This enables the method to predict non-linear capacity degradations and build an explicitly capacity-oriented RUL predictor. Additionally, feature selection based on the random forest (RF) technique was used to enhance the prediction precision. The analyses were conducted based on four separate cycling life testing datasets of a lithium-ion battery. The experimental results indicate that the average percentage of root mean square error for the proposed method is about 2% which respectively is 1.34 times and 8.32 times superior to the LSTM and SVM methods. The outcome of this work can be used for managing the Li-ion battery's improvement and optimization.

KEYWORDS:

Lithium-ion battery, Remaining useful life, Feature engineering, Multivariate time series, Gated recurrent unit.

2. The paper structure is too short and must be elaborated in the technology they applied as well support more rigorous technical aspects. Even though, it is essential to address their method using the algorithm which makes it clear to grasp the steps of the improvements of the method.

Response: Thanks very much for your suggestion. We have made a complete scan on the recent published works in the field of remaining useful life prediction of Li-ion battery, and improved the literature review accordingly.

3. The time and space complexity and algorithm not specified.

Response: Thanks for your valuable comment. Based on your comment, we added executed time for all methods which is listed in Table 4.

TABLE 4 The average of RMSE prediction accuracy and executed time for all cases.

	SVM	LSTM	GRU
RMSE (%)	17.43	2.82	2.09
Executed time (s)	0.9511	19.2397	14.1173

The new paragraph reads as follows:

In addition, to highlight the proposed method, the average of RMSE percentage for all cases has been listed in Table 4. This table shows the GRU-RNN error is almost 2% which is more accurate compared to its peer and appropriate for real-word system. Moreover, the executed time for GRU-RNN is almost 14 seconds which demonstrates that due to fewer parameters, the learning speed is faster than the LSTM with 19 seconds. However SVM has a high execution speed owing to its very simple structure, but it is not a suitable option for time series problems and has very weak prediction accuracy.

4. Test Setup and tuning for the work is expected to elaborate and detailed for the future productions.

Response: Thanks very much for your constructive comment. We had specified the test setup in the previous manuscript.

4 | RESULTS AND DISCUSSION

Matlab 2019 performed data preprocessing and feature engineering in this work, and Python 3.6 was also used for model simulation and training. The simulation was performed on a laptop with a graphic card NVIDIA GeForce 930M at 6 GB, 64-bit operating system and an Intel Core *i*7 – 6500U processor (6 MB cache, up to 3.18 GHz), x64-based processor.

For a fair comparison, model parameters (such as learning rate=1e-5, lag=8, epoch=1000) are considered the same values. Besides, the batch size tuning has been done by a common method, which is called the grid search method. We set an early stopping for both models if the validation accuracy does not increase for 1000 global steps; the training will stop. It is worth noting that, due to the use of the *reduceLR* technique, the impact of the learning rate has diminished, and it has been no need for exact tuning.

5. The literature has to be strongly updated with some relevant and recent papers focused on the fields dealt with the manuscript.

A survey of deep learning techniques: application in wind and solar energy resources." IEEE Access 7 (2019): 164650-164666.

A deep learning ensemble approach for diabetic retinopathy detection. IEEE Access, 7, pp.150530-150539.

Response: Two references are valuable to support this opinion as mentioned. They seem relevant.

****

Reviewer: 2

Comments to the Author

The presented idea is interesting, and the simulations validate the effectiveness of the proposed scheme. However, there are some issues to be improved:

Response: We would like to sincerely thank the reviewer for your time and helpful comments. We have gladly addressed the review comments to further improve our paper.

1- It is strongly recommended that the authors check English grammar and vocabulary. A Professional English editing service might be a good choice.

Response: Thank you for pointing out this issue in our manuscript. We checked again the manuscript and modified the content in terms of grammar and suitable words.

2- The contributions of the paper are not highlighted clearly.

Response: Thanks very much for your constructive comment. The contributions were already outlined in the last paragraph of the introduction. However, we recognize that these contributions could be further clarified and therefore have changed the end of the introduction to emphasize further how our work stands out in comparison to existing work. The new paragraph reads as follows:

battery capacity and observable variables. In particular, the GRU-RNN is an enhanced type of simple RNN to resolve a short-term dependence problem. Compared to the equivalent circuit models and electrochemical models that encompass differential equations, the GRU-RNN doesn't need to extract the battery's internal parameters and many other tasks to parametrization. The key contributions of the present work were summarized as follows:

1. Since time series data are time-dependent, there is a need to consider time delays (lag) to predict such cases that have not been addressed in most papers in battery degradation prediction. Toward this end, a deep learning method is introduced for multivariate time-series prediction.
2. A gated recurrent unit model is proposed with multivariate input to predict the battery RUL. Unlike the LSTM, this method has fewer parameters due to merging the cell state and the hidden state and does not need a memory unit. Therefore, it makes a more straightforward structure and fast training.
3. Feature extraction through the statistical equations and feature selection is done based on random forest. At this stage, not only the computational burden of modeling is reduced, but also the performance of the model is improved.
4. An adaptive learning rate optimization algorithm, namely Adam Optimization, is applied for the GRU-RNN model to optimize the training network. This technique cannot only complete the model training rapidly and stably but also reduce the effect of learning rate and the training time. Besides, an early stop technique is used to prevent overfitting.
5. The investigations on a reliable battery dataset from NASA demonstrate that GRU-RNN can obtain greater precision than the LSTM and traditional methods.

3- In the introduction section, the literature review should be expanded. Due to that, the main contributions of the paper should be described with respect to the references. Additionally, the author should clarify about the complexities found in the previous proposed methods as stated in the last paragraph of the introduction section.

Response: Many thanks for your insightful comment. We have made a complete scan on the recent published works in the field of remaining useful life prediction of Li-ion battery, and improved the literature review accordingly. Please refer to the highlighted sentences in the new introduction section.

4- In order to make the conclusion section more clear, authors need to include the point-by-point findings of this article. The current conclusion is written very wide and it is not easy to maintain the key findings.

Response: Thanks very much for your comment. We have rewritten the conclusion and have specified the work done step-by-step. Now, we hope the revised manuscript could meet the reviewer's requirement. The new paragraph reads as follows:

5 | CONCLUSION

As a crucial tool for prognostic and health management (PHM), the remaining useful life (RUL) prediction is capable of ensuring a possible Li-ion battery failure time in advance. One of the most crucial concerns in the RUL prediction of Li-ion batteries is the way of appropriately learning the long-term dependencies of several hundred cycles while limited degradation data are available.

This paper has been presented for a data-driven model to monitor battery health. The gated recurrent unit (GRU) recurrent neural network (RNN) has been used to predict the battery RUL. To achieve high accuracy prediction, important features based on Pearson correlation and random forest (RF) algorithm have been applied to feed into the GRU-RNN as a multivariate input. Moreover, to optimize the training network, the Adam technique has been applied for convex optimization, which requires low memory. At the same time, an early stopping technique has been used to deal with overfitting and leads to enhance the performance of the GRU-RNN model.

For the experimental and evaluation of our proposed method, the NASA Li-ion battery data set has been applied. The findings have been compared with its sibling technique, which is called LSTM. The results highlight the proposed method has higher accuracy and efficiency than LSTM-RNN and SVM.

****

Reviewer: 3

Comments to the Author

Battery remaining useful life prediction is important for battery management but has been researched many years. Some comments are given as follows:

Response: We would like to sincerely thank the reviewer for your time and helpful comments. We have gladly addressed the review comments to further improve our paper.

1- The introduction part is too simple and should be improved with more surveys of battery lifetime models. The literature review is insufficient. RUL prediction has been widely reported in many journals. Machine learning based methods only belong to one kind of methods.

Response: Thanks a lot for your insightful comment. Upon your suggestion, we have made a complete scan on the recent published works in the field of remaining useful life prediction of Li-ion battery, and improved the literature review accordingly. Please refer to the highlighted sentences in the introduction section of the new manuscript.

2- The motivation of using gated recurrent unit (GRU) rather than LSTM must be described in depth with evidence.

Response: Thank you very much for the thoughtful comment. Both LSTM and GRU are efficient at addressing the problem of vanishing gradient that occurs in long sequence models. However, LSTM-based RNN models have attained state-of-the-art accomplishment on manifold machine learning tasks, and their gating mechanism leads to significant complexity. Compared to LSTM, GRU's have significantly fewer tensor operations and are speedier to train than LSTMs', thus making it a very suitable candidate for embedded implementations [1]. Moreover, the GRU-RNN model can rapidly learn its own parameters by means of an ensemble optimization method based on the Adam optimizers.

[1] Chung, Junyoung, et al. "Empirical evaluation of gated recurrent neural networks on sequence modeling." arXiv preprint arXiv:1412.3555 (2014).

Accordingly, we added some sentences at the end of the introduction to highlight the GRU method rather than LSTM. The new paragraph reads as follows:

However LSTM-based RNN models have attained state-of-the-art accomplishment on various machine learning tasks, the gating mechanism leads to significant complexity. As an alternative, the GRU architecture is similar to the LSTM architecture but has one fewer gate. Compared with an LSTM-based model, a GRU-based model due to the merging of the cell state and the hidden state has a more straightforward structure and fewer tensor operations (about 25% fewer), thus making model training easier and making it a very appropriate candidate for embedded implementations. So far, a few works have been done using GRU methodology in battery RUL prediction, but there are still weaknesses[26,27]. For instance, Song et al.[26] proposed this

And also we have written the motivation items and specified the performance of GRU in item 2. The new paragraph reads as follows:

battery capacity and observable variables. In particular, the GRU-RNN is an enhanced type of simple RNN to resolve a short-term dependence problem. Compared to the equivalent circuit models and electrochemical models that encompass differential equations, the GRU-RNN doesn't need to extract the battery's internal parameters and many other tasks to parametrization. The key contributions of the present work were summarized as follows:

1. Since time series data are time-dependent, there is a need to consider time delays (lag) to predict such cases that have not been addressed in most papers in battery degradation prediction. Toward this end, a deep learning method is introduced for multivariate time-series prediction.
2. A gated recurrent unit model is proposed with multivariate input to predict the battery RUL. Unlike the LSTM, this method has fewer parameters due to merging the cell state and the hidden state and does not need a memory unit. Therefore, it makes a more straightforward structure and fast training.
3. Feature extraction through the statistical equations and feature selection is done based on random forest. At this stage, not only the computational burden of modeling is reduced, but also the performance of the model is improved.
4. An adaptive learning rate optimization algorithm, namely Adam Optimization, is applied for the GRU-RNN model to optimize the training network. This technique cannot only complete the model training rapidly and stably but also reduce the effect of learning rate and the training time. Besides, an early stop technique is used to prevent overfitting.
5. The investigations on a reliable battery dataset from NASA demonstrate that GRU-RNN can obtain greater precision than the LSTM and traditional methods.

3- When describe the battery RUL and capacity prediction, please consider these highly related works: Gaussian process regression with automatic relevance determination kernel for calendar aging prediction of lithium-ion batteries; Modified Gaussian process regression models for cyclic capacity prediction of lithium-ion batteries; An evaluation study of different modelling techniques for calendar ageing prediction of lithium-ion batteries.

Response: Thanks for your suggestion. Yes, Gaussian process regression (GPR) is a kind of Bayesian nonparametric model and a powerful method for prediction. With the development of machine learning, GPR is receiving much attention because the algorithm can provide a principled, practical, and probabilistic approach and can give an easy framework for modeling and prediction. Compared with other neural network algorithms, which need many decision parameters such as activation function and learning rate, the GPR has concise parameters to build a model that more natural to handle and understand the learning process clearly. Accordingly, we have cited these three interesting papers in the introduction section, and also have discussed GPR in the section. The new paragraph reads as follows:

vant information long before the current state. However, there is still a problem of poor prediction ability using the data-driven approaches for battery health prediction owing to the nonlinear structure of the LiBs.

Compared with other data-driven methods, the Gaussian process regression (GPR) method is a class of Bayesian model that has strong nonlinear modeling capability to solve and predict the regression problems [20][21][22]. Moreover, the GPR method can also improve prediction accuracy without the physical model. However, the trend fitting deteriorates when test data is far from the training data, and the predictive results are unsatisfactory. In this regard, some attractive papers have been published for calendar aging prediction of Li-Ion batteries using the modified GPR method. For instance, a mechanism-conscious GPR model has been constructed for battery cycle life prediction. In this way, by coupling the polynomial equation and Arrhenius law into a compositional kernel through the GPR model, a modified model of the GPR was made, which could have an acceptable prediction against uncertainties[23]. In other interesting work, an advanced Gaussian filter technique has been performed to obtain the smoothing incremental capacity curves. Then the health indexes (HIs) have been extracted from the partial incremental capacity

curves as the input features of the GPR model. The results were demonstrated that the proposed model has advantages of high accuracy and robustness[24].

4- The measurement and shift noises from sensors would highly affect your RUL prediction method. Have you considered the effects of these noises?

Response: Thanks for your valuable comments. This is a very insightful question. We agree with your opinion that the noises from sensors would highly effect on RUL perdition and state estimation. Due to the fact that the NASA dataset also includes noise, we didn't include any noise or disturbance separately. If I want to explain how the proposed method works well against the noise, I need to explain some things. Please see the figure below:

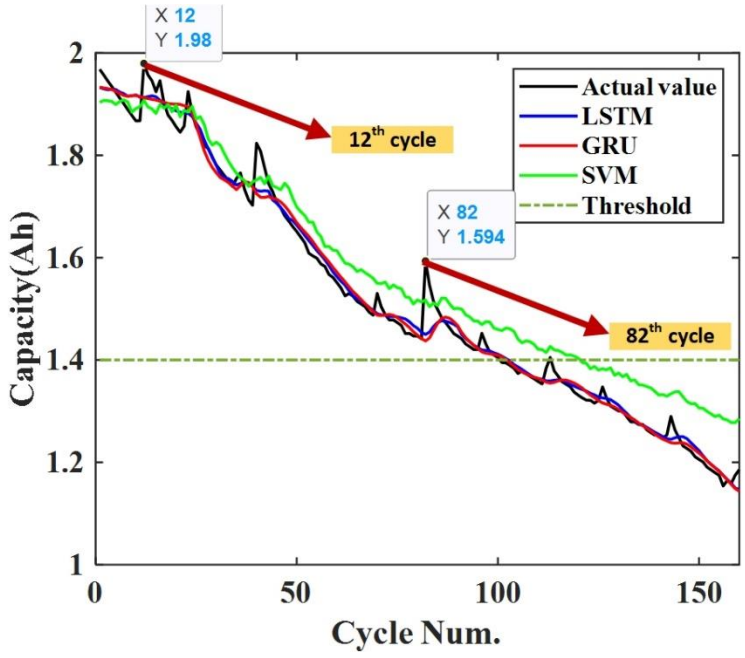

Fig.1: Capacity estimation for Battery #6.

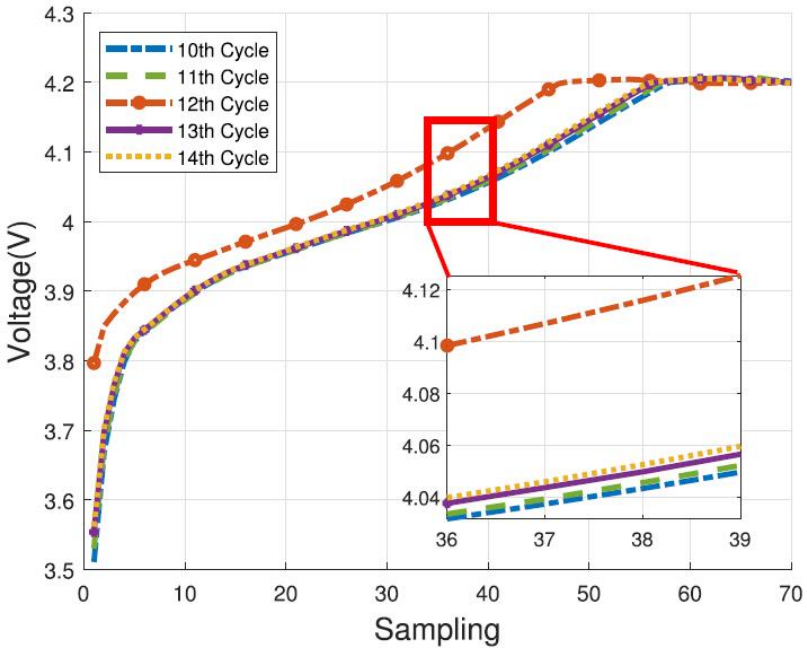

Fig.2: Abnormal voltage pattern at 12th cycle of Battery #6.

In the above figure (Fig.2), we observe some glitches around 12th or 82th cycles; even though the true capacity is not severely low, most learning methods estimate the capacity very low in these cycles. This is because the voltage charging profile at 12th cycle is abnormally deviated from those of other cycles as shown in the figure below. We also observe similar deviations in all four battery charging profiles of current and temperature; these abnormal profiles may be due to measurement noise or sensory misfunction. Consequently, the estimated capacity is far from the true capacity.

However, unlike the other existing methods, GRU-RNN estimates the capacity accurately even at the 12th cycle, as shown in the Fig.3. This is because the structure of GRU-RNN considers the long-term information, and this leads to suppress the effect of weight at the 12th cycle data. It is noteworthy that not only GRU has this capability but also LSTM has such performance against the noise, such as mentioned in the below two references [1, 2]. Accordingly, we have discussed this issue in this manuscript.

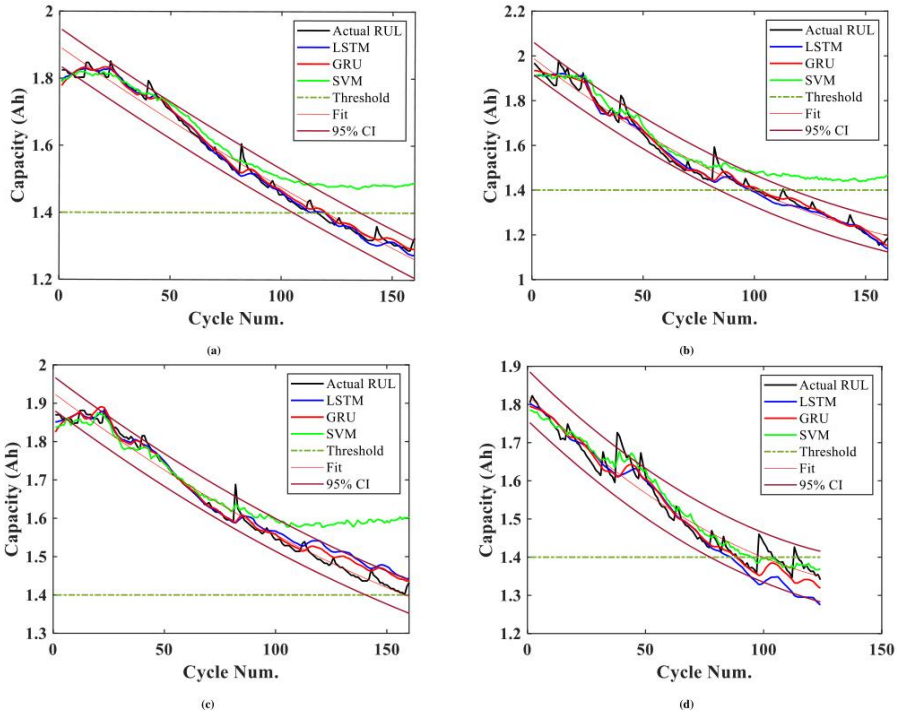

FIGURE 9 RUL prediction (60% of training); (a) B0005, (b) B0006, (c) B0007, (d) B0018.

Fig.3: RUL prediction.

[1] Choi, Y., Ryu, S., Park, K. and Kim, H., 2019. Machine learning-based lithium-ion battery capacity estimation exploiting multi-channel charging profiles. IEEE Access, 7, pp.75143-75152.
[2] Zhang, Y., Xiong, R., He, H. and Pecht, M.G., 2018. Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries. IEEE Transactions on Vehicular Technology, 67(7), pp.5695-5705.

The new paragraph reads as follows:

- **Reset gate**: The reset gate controls the level of ignorance of information in h_{t-1}. If the value of the reset gate is small, the information is more overlooked. This parameter can be applied for the prediction of RUL of lithium-ion batteries for

rejecting outliers, noises, and unnecessary degradation information between adjacent cycles[29]. This is because the structure of RNN-based model which considers long-term information, and this leads to suppress the effect of weight at the adjacent cycle data.[30]. Eq. (5) and Eq. (6) reflect the GRU-RNN reset operation.

It is worth noting that, in future work, we are going to add a section for analysis of robustness against noise and disturbance as uncertainties. Thanks again for your valuable comment.

5- It is good that you use the random forest to select importance feature. Have you use it to quantify the correlations among these features.

Response: We understand that this question meant to confirm “Have you selected the important features using RF?”

In this study, the first step we used statistical features for extracting the features from raw data (voltage, Current, and Temperature). Then using random forest we calculated the correlation between features and capacity. This means that the numerical value of the correlation of each feature and capacity is calculated and then ranked, and only 15-top features that have a stronger correlation with capacity are selected. This is done to reduce the dimension of the features for network input and decrease the computation. And the random forest also eliminates the features that are less important. For clarification purposes, Fig. 7 (in updated manuscript) has been improved and now it shows the heatmap associated with the quantity amount.

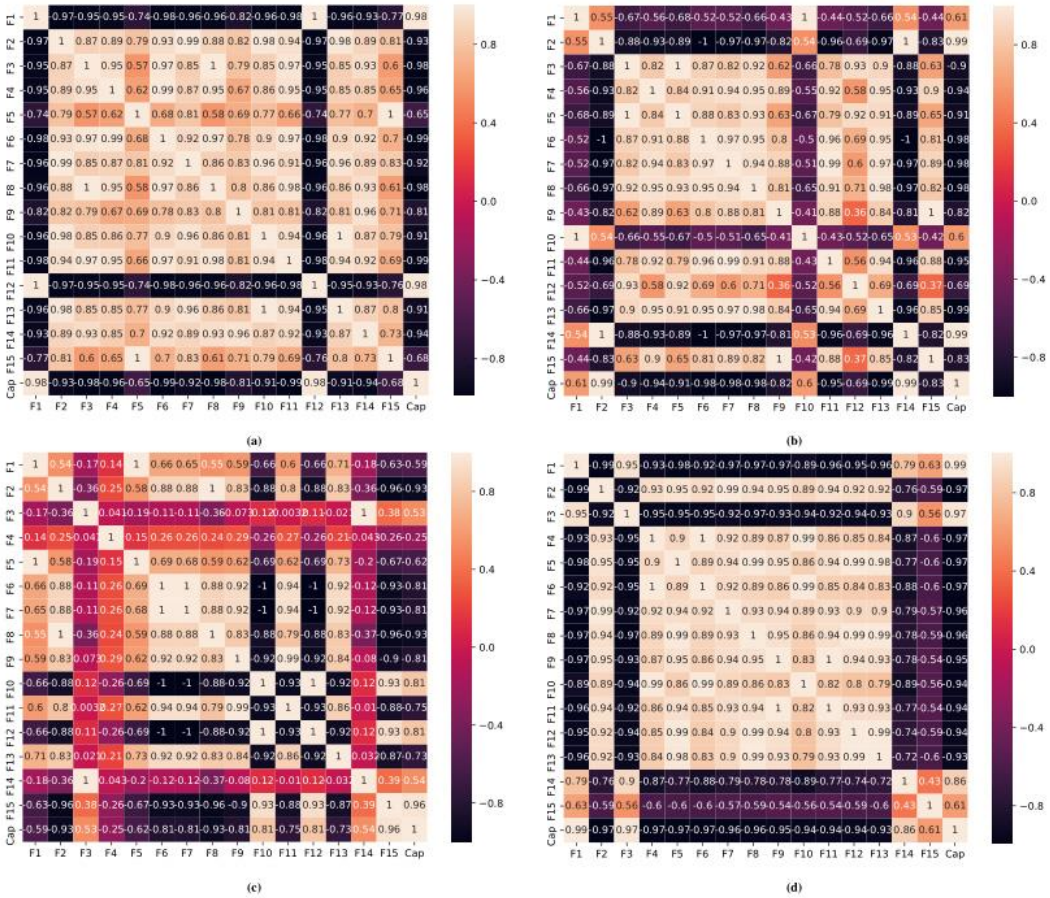

FIGURE 7 Heatmap of selected features using RF algorithm; (a) B0005, (b) B0006, (c) B0007, (d) B0018.

6- RF and GPR are also powerful tool to quantify feature correlations, please clarify it with recommended work: Feature Analyses and Modelling of Lithium-ion Batteries Manufacturing based on Random Forest Classification.

Response: We appreciate your insightful suggestions for the relevant paper. We have added this reference to the manuscript, since it seems relevant. The new sentences reads as follows:

methods such as random forest (RF), normalization cross-correlation indicator method, isometric mapping (ISOMAP) method, and other techniques are suggested to model the degradation process to achieve a more discriminative feature space. In this regard, a comprehensive paper[12] has been published, which proposed a framework based on random forest classification for lithium-ion battery feature analysis. The results illustrated that the random forest technique attains the reliable classification of battery physical properties and leads to the impressive quantification of both correlations and feature importance.

12. Liu Kailong, Hu Xiaosong, Zhou Huiyu, Tong Lei, Widanalage Dhammika, Macro James. Feature Analyses and Modelling of Lithium-ion Batteries Manufacturing based on Random Forest Classification. *IEEE/ASME Transactions on Mechatronics*. 2021;.

7- Uncertainty quantification is also important for RUL prediction, have you considered this? Some effective solution such as using the hybrid model with GPR and LSTM can achieve this. Please give some comments.

Response: Thank you for spotting this. It is an important question. We agree with your opinion that the uncertainty quantification would highly effect on RUL perdition and state estimation. This question seems to be along with your previous comment (Comment #4). Based on your comment, we have added 95% confidence interval (CI) as uncertainty quantification. In figure below, it show our proposed method is between two boundaries (lower bound and

upper bound) and reflect the reliably of the RUL prediction for proposed model. We hope this is now clear based on our answer to your nice comment. The new sentences read as follows:

where $Y(i)$ and $\hat{Y}(i)$ denoted to the predicted capacity and measurement capacity series, respectively. i is the number of cycles between the actual battery and first prediction cycle. Besides, the 95% confidence interval (CI) of all methods are performed for evaluation of the uncertainty as

$$95\%CI = \hat{\bar{Y}}(i) \pm 1.96 \times \sigma^2(\hat{Y}(i)) \tag{20}$$

where 95%CI is the confidence interval for RUL prediction. $\hat{\bar{Y}}(i)$ and σ^2 denote mean values of RUL prediction and variance of the predicted values, respectively.

And also, we have updated the results and exhibited the lower bound and upper bound to assess the uncertainty. The new results read as follows

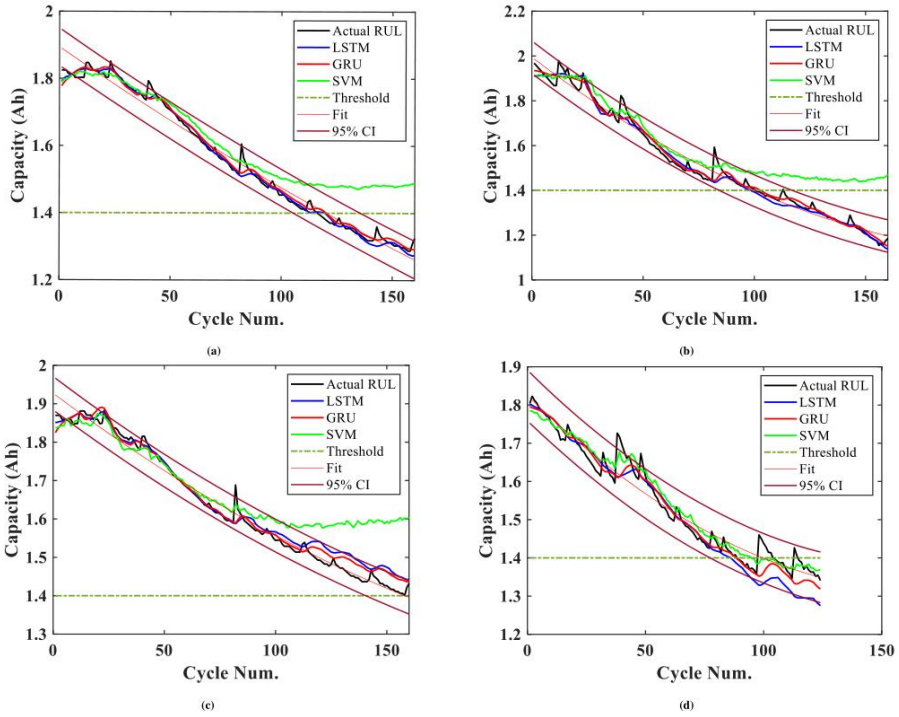

FIGURE 9 RUL prediction (60% of training); (a) B0005, (b) B0006, (c) B0007, (d) B0018.

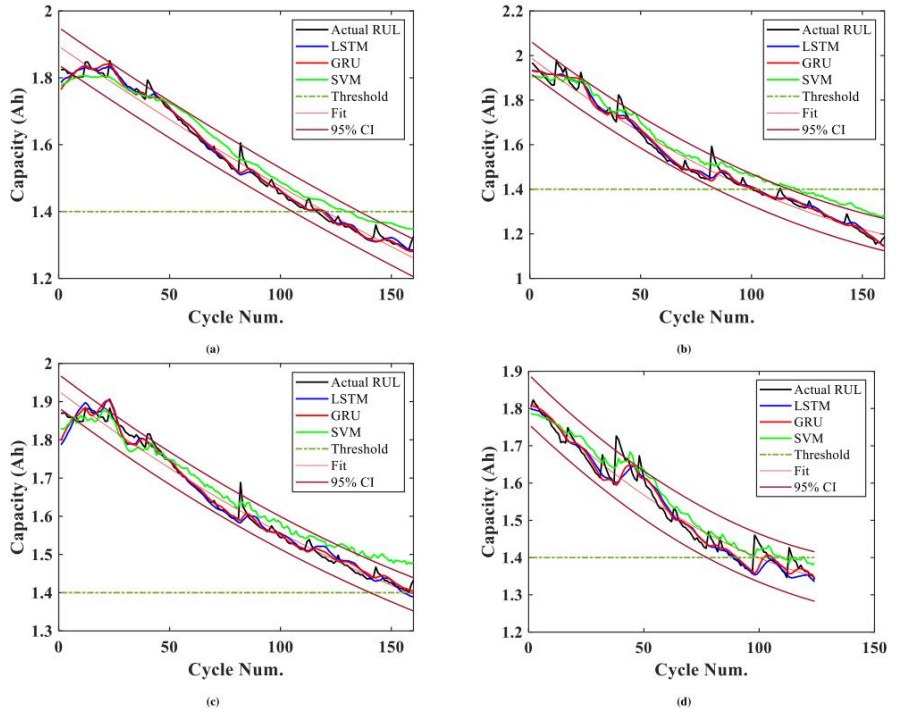

FIGURE 12 RUL prediction (80% of training); (a) B0005, (b) B0006, (c) B0007, (d) B0018.

As to the second part of your comment, we investigated the approach you mentioned. We figured out that the combination of LSTM/GRU and GPR as a hybrid model can be a good option to predict the battery RUL and analyze its robustness against the uncertainty. As we know, Gaussian process regression (GPR) is a method for generating probabilistic data in the process of prediction. The GPR adopts the kernel function based on Bayes linear regression, which increases its ability to deal with intricate and nonlinear problems. Furthermore, it has good adaptability and strong generality to process complex nonlinear problems. On the other hand, the LSTM/GRU network is suitable for dealing with important events with longer intervals and delays in time series. Therefore the combination of these two powerful methods as a hybrid model not only can predict the long-term information, but also can be robust against the uncertainties in the harsh environment [1]. However, the combination of these two methods (GPR and LSTM) increases the computational burden and cost. For this reason, in this study, we intended to use the GRU/LSTM separately so that it has a less computational burden and is easier to implement in real-world applications.

[1] Wang, Y., Feng, B., Hua, Q.S. and Sun, L., 2021. Short-Term Solar Power Forecasting: A Combined Long Short-Term Memory and Gaussian Process Regression Method. Sustainability, 13(7), p.3665.

8- The description in conclusion part is confusing and it is suggested to rewrite it.

Response: Thanks very much for your comment. We have rewritten the main paragraph of the conclusion and have specified the work done step-by-step. Now, we hope the revised manuscript could meet the reviewer's requirement. The new paragraph reads as follows:

5 | CONCLUSION

As a crucial tool for prognostic and health management (PHM), the remaining useful life (RUL) prediction is capable of ensuring a possible Li-ion battery failure time in advance. One of the most crucial concerns in the RUL prediction of Li-ion batteries is the way of appropriately learning the long-term dependencies of several hundred cycles while limited degradation data are available.

This paper has been presented for a data-driven model to monitor battery health. The gated recurrent unit (GRU) recurrent neural network (RNN) has been used to predict the battery RUL. To achieve high accuracy prediction, important features based on Pearson correlation and random forest (RF) algorithm have been applied to feed into the GRU-RNN as a multivariate input. Moreover, to optimize the training network, the Adam technique has been applied for convex optimization, which requires low memory. At the same time, an early stopping technique has been used to deal with overfitting and leads to enhance the performance of the GRU-RNN model.

For the experimental and evaluation of our proposed method, the NASA Li-ion battery data set has been applied. The findings have been compared with its sibling technique, which is called LSTM. The results highlight the proposed method has higher accuracy and efficiency than LSTM-RNN and SVM.

****

Reviewer: 4

Comments to the Author

This manuscript presents a GRU-RNN based battery Remaining Useful Life prediction approach, and compared the prediction results with LSTM and SVM method using four separate cycling life testing datasets from NASA. The manuscript is well organized and the adequate literature is cited. However, there are some uncertainties need to be addressed before its publication. Please consider the following comments:

Response: The authors sincerely appreciate the reviewer's positive and constructive comments. Following the review comments, we have carefully modified the manuscript, as detailed in the following.

1- The authors are suggested to give more detailed explanation for the feature parameter of the 30 time-domain features, which seems extract from battery signals based on statistical equations.

Response: As further clarification, we have added a table of statistical formulas which we used for time-domain feature extraction (Please see Table 1). Moreover, we improved the heatmap of selected features using the RF algorithm. In this figure, we demonstrated the quantification of the feature correlation.

TABLE 1 Statistical formulas in the time-domain.

Name	Formula	Name	Formula	Name	Formula
Mean	$F_m = \frac{1}{N}\sum_i^N x(i)$	Shape Factor (SHF)	$F_{shf} = \frac{F_{rms}}{\frac{1}{N}\sum_i^N \lvert x(i)\rvert}$	Skewness Factor (SF)	$F_{sf} = \frac{\frac{1}{N}\sum_i^N \lvert x(i)\rvert^3}{F_{rms}^{\ 3}}$
Standard Deviation (STD)	$F_{std} = \sqrt{\frac{1}{N}\sum_i^N (x(i)-\overline{x})}$	Crest Factor (CF)	$F_{cf} = \frac{F_p}{F_{rms}}$	Kurtosis Factor (KF)	$F_{kf} = \frac{\frac{1}{N}\sum_i^N \lvert x(i)\rvert^4}{F_{rms}^{\ 4}}$
Root Mean Square (RMS)	$F_{rms} = \sqrt{\frac{1}{N}\sum_i^N x(i)^2}$	Impulse Factor (IF)	$F_{if} = \frac{F_p}{\frac{1}{N}\sum_i^N \lvert x(i)\rvert}$	Clearance Factor (CF)	$F_{clf} = \frac{F_p}{\frac{1}{N}\sum_i^N \sqrt{\lvert x(i)\rvert}}$
Peak	$F_p = \max \lvert x(i)\rvert$	–	–	–	–

$x(i)$ is the battery signals series; $\overline{x}$ is the mean value of the series.

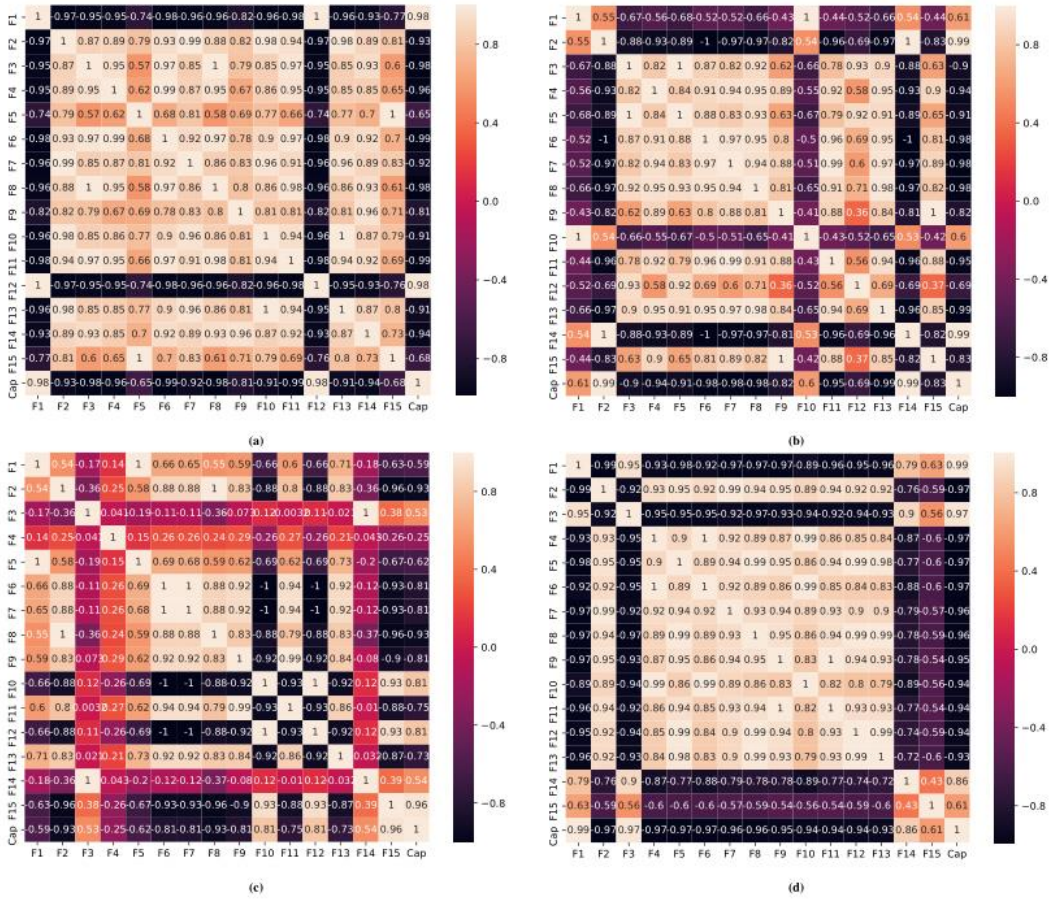

FIGURE 7 Heatmap of selected features using RF algorithm; (a) B0005, (b) B0006, (c) B0007, (d) B0018.

2- The time-domain features are extracted from the battery signals during the entire discharge process, while the

battery is not fully (dis)charged in daily applications. In this case, how to verify the proposed time-domain features?

Response: Thank you very much for the insightful comments. This is a very good point. Yes, it is one of the major challenges in real-world applications.

In this paper, we have tried to extract features that can be predicted in case the battery is not fully discharged. Allow me to explain with an example through the below figure, which shows how time-domain features can be used in this issue. As you can see in the figure below, suppose the battery is not fully charged and its voltage level is 3.725 volts. Also, the discharge is not complete and drops to the voltage level of 3.38 volts. That is, only the blue line shows the battery cycle (does not include the red dashed lines). Now suppose we have the same cycle for current. In such cases, the discharge time is from 40 mins to 160 mins. Suppose that during this period, 7200 points of voltage and current signals are sampled. (120 mins * 60 s = 7200 samples, i.e. 1 sample per second). Then, using statistical formulas, various time-domain features can be extracted. For example, the below two time-domain features can be extracted from voltage and current data from the incomplete cycle (namely between 40-160 mins):

$$V_{RMS} = \sqrt{\frac{1}{N}\sum_{i}^{N} V(i)^2} = \sqrt{\frac{1}{7200}(v_1^2 + v_2^2 + v_3^2 + \ldots + v_{7200}^2)} = \text{Constant value}$$

$$I_{Mean} = \frac{1}{N}\sum_{i}^{N} I(i) = \frac{1}{7200}(I_1 + I_2 + I_3 + \ldots + I_{7200}) = \text{Constant value}$$

Where N is the number of samples in a cycle.

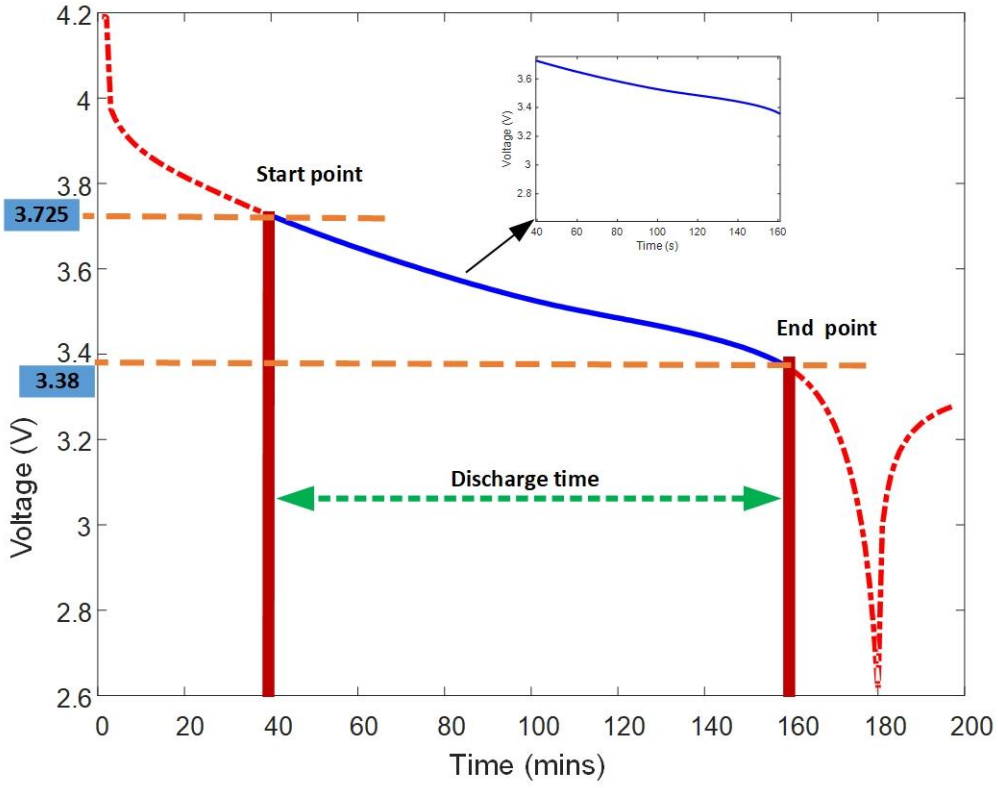

Fig: Discharge voltage

3- The MAE, RMSE and RUL error of prediction based on SVM model are provided in the tables and boxplot figures, but there are no capacity prediction results in Figure 9/11/12/14. The authors are recommended to refer

the latest publications such as Journal of Power Sources 421, 56-67; Energy 190, 116467.

Response: Thank you for spotting this. At the beginning, we plot the SVM model prediction in results. But the SVM model prediction was not very accurate for prediction with 60% of training data. Therefore, we didn't show the SVM curves. Based on your comment, we added the curves again and also increased the resolution of the all figures.
As to the second part of your comment, we have made a complete scan on the recent published works in the field of remaining useful life prediction of Li-ion battery, and improved the literature review accordingly. The new paragraph reads as follows:

Compared with other data-driven methods, the Gaussian process regression (GPR) method is a class of Bayesian model that has strong nonlinear modeling capability to solve and predict the regression problems [20,21,22]. Moreover, the GPR method can also improve the accuracy of prediction without the physical model. However, the trend fitting deteriorates when test data is far from the training data and the predictive results are unsatisfactory. In this regard, some attractive papers for calendar aging prediction of Li-Ion batteries using the modified GPR method have been published. For instance, a mechanism-conscious GPR model has been constructed for battery cycle life prediction. In this way, by coupling the polynomial equation and Arrhenius law into a compositional kernel through GPR model, a modified model of the GPR was made, which was able to have an acceptable prediction against uncertainties[23]. In other interesting work, an advanced Gaussian filter technique has been performed to

obtain the smoothing incremental capacity curves. Then the health indexes (HIs) have been extracted from the partial incremental capacity curves as the input features of the GPR model. The results was demonstrated that the proposed model has advantages of high accuracy and robustness[24].

22. Li Xiaoyu, Wang Zhenpo, Yan Jinying. Prognostic health condition for lithium battery using the partial incremental capacity and Gaussian process regression. *Journal of power sources.* 2019;421:56–67.

24. Li Xiaoyu, Yuan Changgui, Li Xiaohui, Wang Zhenpo. State of health estimation for Li-Ion battery using incremental capacity analysis and Gaussian process regression. *Energy.* 2020;190:116467.

4- In section 2.1, the SoC(t) is defined as the measured battery capacity at cycle t in this manuscript. But SoC usually means the real-time scale state of charge.

Response: Thank you for your attention. Yes, you are right. In our work, we used the available battery capacity in each cycle. To avoid misunderstanding, it has been changed to C(t).

5- Since the number of cycles of B0018 is less than other three batteries, the start point for 60% training data is not 93th. Please carefully read proof the manuscript to avoid minor mistake.

Response: Thanks for noticing this inconsistency. We have corrected this in the result section. The modified sentences reads as follows:

80%. In the first scenario, Fig. 9 is shown the battery RUL prediction with 60% training data for four different cases in which the start point for prediction is from the 97^{th} cycle for B0005, B0006, B0007 and 75^{th} cycle for B0018. Fig. 11 is shown the prediction error for both GRU-RNN and LSTM-RNN as well. For the purpose of indicating the GRU-RNN model's accuracy, a comparison has been made between the GRU-RNN, LSTM-RNN, and SVM methods. Fig. 10 is shown a box plot of all training and testing errors together. For all the cases, GRU-RNN has the lowest error, and the LSTM-RNN has less accuracy than GRU-RNN. Meanwhile, the SVM has a big difference from them, which is not suitable for the long-term dependency prediction. Given that the evaluation of models with MAE, RMSE, and RUL prediction, Table. 2 provided a comparison between the models. For the second scenario, we applied 80% data for training and compared the different methods. Fig. 12 is shown the battery RUL prediction with 80% training data for four different cases, which the start point for prediction is from the 129^{th} cycle for B0005, B0006, B0007 and 100^{th} cycle for B0018. The prediction error is shown in Fig. 14, and also the box plot of all training and testing error illustrates the GRU-RNN has less amplitude of error than the other methods. For better methods comparison, Table. 3 is provided the MAE, RMSE, and RUL prediction error. The results achieved in the above section show that the deep

****

Reviewer: 5 (EiC)

Comments to the Author

- Provide more appealing title with no acronyms in precise and concise manner.
- Omit trivial information.
- Explain how the present paper differs from the published ones.
- State specific objectives.
- Get its English edited very carefully.
- Provide better quality figures.
- State main findings in the conclusions.
- Reduce the content similarity index to less than 15% with no more than 1% from any source.
- Finally, it is reminded that increasing the number of co-authors is NOT allowed.

Response: We would like to sincerely thank the associate editor for your time and guidance to help us improve our paper. Following the comments from the reviewers and associate editor, we have more clearly justified the contribution of this paper, significantly improved the literature review (i.e., section I-introduction, page 1-3), improved the resolution of figures, and updated both the abstract and conclusion. Please refer to the following review responses and the highlighted parts in the newly uploaded manuscript.

Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

RESEARCH ARTICLE

Multivariate Gated Recurrent Unit for Battery Remaining Useful Life Prediction: A Deep Learning Approach

Reza Rouhi Ardeshiri[1] | Chengbin Ma*[1]

[1]University of Michigan-Shanghai Jiao Tong University Joint Institute, Shanghai Jiao Tong University, Shanghai, P. R. China.

Correspondence
*Chengbin Ma, Email: chbma@sjtu.edu.cn

Present Address
University of Michigan-Shanghai Jiao Tong University Joint Institute, Shanghai Jiao Tong University, Shanghai, P. R. China.

Summary

This paper proposes the gated recurrent unit (GRU)-recurrent neural network (RNN), a deep learning approach to predict the remaining useful life (RUL) of Lithium-ion batteries, accurately. The GRU-RNN structure can self-learn the network parameters by utilizing adaptive gradient descent algorithms, leading to a reduced computational cost. Unlike the long short-term memory (LSTM) model, GRU-RNN allows time-series dependencies to be tracked between degraded capacities without using any memory cell. This enables the method to predict non-linear capacity degradations and build an explicitly capacity-oriented RUL predictor. Additionally, feature selection based on the random forest (RF) technique was used to enhance the prediction precision. The analyses were conducted based on four separate cycling life testing datasets of a lithium-ion battery. The experimental results indicate that the average percentage of root mean square error for the proposed method is about 2% which respectively is 1.34 times and 8.32 times superior to the LSTM and SVM methods. The outcome of this work can be used for managing the Li-ion battery's improvement and optimization.

KEYWORDS:
Lithium-ion battery, Remaining useful life, Feature engineering, Multivariate time series, Gated recurrent unit.

1 | INTRODUCTION

Due to the low self-discharge rate of lithium-ion batteries (LiBs), high energy density, and the high working voltage, they become a primary choice of the onboard energy storage system in electric vehicles (EVs) [1,2]. Although, in consumer electronics, mobile devices, and EVs, numerous field failures of LiBs are documented. Battery safety has become a critical problem that needs to be intensively investigated. For complicated operational environments in EVs, maintaining battery packs' stability and protection poses a significant technical challenge. Overheat, overcharge, and short circuit are the principal potential failures in the battery packs in EVs [3]. This happens due to several operating conditions, chemical reactions, and mechanical stress.

Prognostics and health management (PHM) of battery technologies has recently attracted a lot of research interest. The enabling discipline of PHM includes methods as well as technologies for assessing the systems' reliability under real-cycle conditions, for the diagnosis of initiated failures and likely failure prognosis [4]. The lithium-ion battery PHM helps users to make tentative maintenance choices to prevent unexpected failure. As one of the most important states to be tracked in a battery and one of the key approaches for PHM, remaining useful life (RUL) is defined as the remaining number of charge-discharge cycles of the battery before the capacity deteriorates to a predetermined failure threshold.[5]. Hence, the battery remaining useful life

(RUL) monitors the future operating status of the battery to manage the charge-discharge of the battery, prolong the battery life, prevent security risks, and decrease the use costs [6].

1.1 | Literature review

Battery degradation process models are often constructed through traditional prediction of battery RUL approaches based on empirical models or linear model assumptions known as model-driven techniques, which are intended to develop mathematical or physical models for explaining battery degradation mechanism and modify model parameters by employing actual data calculated. For instance, Chen et al. proposed a combination of the particle filter (PF) and sliding-window grey model (SGM) to build a new structure for battery RUL prediction. The proposed method was able to continuously and effectively update model parameters to reflect the changing trend of capacity [7]. In another work, Chang et al. presented a hybrid model in which the unscented Kalman filter (UKF) method was adopted to achieve a prognostic result based on an estimated model and generate a raw error series [8]. However, with a complex nonlinear system, an exact mathematical or physical model is impossible to be constructed. In this context, the effectiveness of data-driven models has been attracted high interest. These models typically make decisions regarding online cloud, or edge terminal data based on considerable historical data [9,10]. Such models generally create particular machine learning-based models with more excellent capability for complex nonlinear interactions.

Thus, data-driven methods usually provide more statistical capabilities for data distributions [11]. This method builds upon artificial neural networks (ANNs) and can be supervised, semi-supervised or unsupervised. It uses non-linear functions to establish a relationship between the input and target parameters and uses specific methods to calculate the function parameters. However, the noise and uncleanness of the data dramatically impact solving problems with deep learning methods. Nevertheless, clean data cannot be obtained without noise for data-driven procedures, and it is challenging to identify robust features among the wide range of features such as time domain, time-frequency domain, and frequency domain. Consequently, new feature selection methods such as random forest (RF), normalization cross-correlation indicator method, isometric mapping (ISOMAP) method, and other techniques are suggested to model the degradation process to achieve a more discriminative feature space. In this regard, a comprehensive paper [12] has been published, which proposed a framework based on random forest classification for lithium-ion battery feature analysis. The results illustrated that the random forest technique attains the reliable classification of battery physical properties and leads to the impressive quantification of both correlations and feature importance.

Recent advancements in deep learning techniques have considerably increased the capacity of complex data analysis [13,14]. Besides, deep learning technology has been designed to overcome the requirement for prediction problems since it is particularly advantageous for extremely complicated nonlinear fittings of artificial neural networks. New challenges for complex prediction problems, including accurate prediction of RUL battery, are expanded by the deep learning platform. Numerous innovative deep learning methods have provided plenty of advantages for battery health monitoring in recent years [15,16].

Numerous research activities concentrated on models based on the deep neural network (DNN), such as the ensemble learning, deep belief network (DBN), and extreme learning machine (ELM). These frameworks are often concentrated on fault measurement areas with less time series knowledge requirements [17]. On the other hand, the convolution neural network (CNN) and RNN networks were proposed to the RUL field in recent years. For example, Li et al. built a novel framework using compact convolutional neural network models through the concepts of transfer learning (TL) to improve the battery health estimation accuracy [18]. Another disadvantage of these methods is that the chosen prediction model possesses more input parameters and need to have time consecutive [19]. It is noteworthy that the typical RNN algorithm suffers from connecting the current input to the relevant information long before the current state. However, there is still a problem of poor prediction ability using the data-driven approaches for battery health prediction owing to the nonlinear structure of the LiBs.

Compared with other data-driven methods, the Gaussian process regression (GPR) method is a class of Bayesian model that has strong nonlinear modeling capability to solve and predict the regression problems [20,21,22]. Moreover, the GPR method can also improve prediction accuracy without the physical model. However, the trend fitting deteriorates when test data is far from the training data, and the predictive results are unsatisfactory. In this regard, some attractive papers have been published for calendar aging prediction of Li-Ion batteries using the modified GPR method. For instance, a mechanism-conscious GPR model has been constructed for battery cycle life prediction. In this way, by coupling the polynomial equation and Arrhenius law into a compositional kernel through the GPR model, a modified model of the GPR was made, which could have an acceptable prediction against uncertainties [23]. In other interesting work, an advanced Gaussian filter technique has been performed to obtain the smoothing incremental capacity curves. Then the health indexes (HIs) have been extracted from the partial incremental capacity

curves as the input features of the GPR model. The results were demonstrated that the proposed model has advantages of high accuracy and robustness[24].

On the other hand, LSTM-RNN is another class of RNN that could solve some drawbacks of simple RNN, such as vanishing and exploding gradient[25]. Li et al implemented an architecture using LSTM-based time series processing, which allows the input charging curves to be variable in time steps and prediction to be attained even with incomplete sensor data[25]. Although, LSTM's configuration is complex and is composed of three gates, including the output gate, input gate, and forget gate.

However LSTM-based RNN models have attained state-of-the-art accomplishment on various machine learning tasks, the gating mechanism leads to significant complexity. As an alternative, the GRU architecture is similar to the LSTM architecture but has one fewer gate. Compared with an LSTM-based model, a GRU-based model due to the merging of the cell state and the hidden state has a more straightforward structure and fewer tensor operations (about 25% fewer), thus making model training easier and making it a very appropriate candidate for embedded implementations. So far, a few works have been done using GRU methodology in battery RUL prediction, but there are still weaknesses[26,27]. For instance, Song et al.[26] proposed this method to predict battery degradation. However, this study has not been considered the battery features and just applied capacity observation as input and has not been used multivariate time series prediction. In other work, Ungurean et al.[27] proposed online state of health (SOH) estimation for LiBs using GRU. In the first step, they estimated the state of charge (SOC) and then used battery capacity to predict SOH. However, estimation of SOC and then using it to predict SOH is complex and will result in many errors. This method will be involved in two predictions that if the SOC estimation error is high, the SOH estimation error will be more. It is worth noting that they used univariate GRU for prediction. In this paper, a data-driven precise battery RUL prediction model is developed using an effective training network to cope with these limitations.

1.2 | Motivations and contributions

A Gated recurrent unit is a modern deep learning network to overcome the abovementioned problems. For verifying the effects, the comparison of a number of state-of-the-art models with the suggested GRU in the present study is performed. Dissimilar to the above RNN-based approaches, and the GRU-RNN-based RUL approach is proposed to create nonlinear mapping among the battery capacity and observable variables. In particular, the GRU-RNN is an enhanced type of simple RNN to resolve a short-term dependence problem. Compared to the equivalent circuit models and electrochemical models that encompass differential equations, the GRU-RNN doesn't need to extract the battery's internal parameters and many other tasks to parametrization. The key contributions of the present work were summarized as follows:

1. Since time series data are time-dependent, there is a need to consider time delays (lag) to predict such cases that have not been addressed in most papers in battery degradation prediction. Toward this end, a deep learning method is introduced for multivariate time-series prediction.
2. A gated recurrent unit model is proposed with multivariate input to predict the battery RUL. Unlike the LSTM, this method has fewer parameters due to merging the cell state and the hidden state and does not need a memory unit. Therefore, it makes a more straightforward structure and fast training.
3. Feature extraction through the statistical equations and feature selection is done based on random forest. At this stage, not only the computational burden of modeling is reduced, but also the performance of the model is improved.
4. An adaptive learning rate optimization algorithm, namely Adam Optimization, is applied for the GRU-RNN model to optimize the training network. This technique cannot only complete the model training rapidly and stably but also reduce the effect of learning rate and the training time. Besides, an early stop technique is used to prevent overfitting.
5. The investigations on a reliable battery dataset from NASA demonstrate that GRU-RNN can obtain greater precision than the LSTM and traditional methods.

1.3 | Organization of the paper

The rest of the article is structured as the following: Section 2 outlines the history of the proposed method as well as its architecture, Section 3 discusses the new approach for RUL prediction related to model optimization and feature engineering in detail. Section 4 is about the results of RUL prediction and discussion, and finally, Section 5 summarizes the conclusions.

2 | RELATED WORK

2.1 | RNN architecture

RNN is a deep neural network class applied to assess time dependencies and input features on a sequential input to predict future output, with particular characteristics called internal cell state or memory. Therefore, each neuron's output varies depending on the current input and the background of previously hidden state outputs. Fig. 1 indicates the function of the unfolded RNN structure with a feedback loop on a simple RNN, which can retain background information efficiently according to the number of time steps[28].

The assumption is that the input and output vectors of the RNN are $\mathbf{X}(t) = \{x_0, x_1, ..., x_t\}$ and $\mathbf{Y}(t) = \{y_0, y_1, ..., y_t\}$, respectively, which can have arbitrary dimensions. Hence, the battery data set applied to train the model can be illustrated as the following:

$$\Psi = \{X, Y\} \tag{1}$$

Here, $x_t = [I, V, T]$ and $y_t = [C(t)]$, where

$$\begin{cases} I = [I_1, ..., I_m], & m = \text{number of current-related features.} \\ V = [V_1, ..., I_n], & n = \text{number of voltage-related features.} \\ T = [T_1, ..., I_p], & p = \text{number of temperature-related features.} \end{cases}$$

and $C(t)$ is the measured battery capacity at cycle t.

RNNs operate by iterative update of a hidden state, h, that is also a vector with arbitrary dimensions. First, at any step t, the next hidden state h_t is determined with the next input x_t and the hidden state h_{t-1}. Secondly, h_t is used to measure the next y_t output. The equations that mathematically define a single RNN cell in a single-layer RNN can be seen Eq. (2) and Eq. (3) as follows:

$$h_t = \tanh(W_{hx}x_t + W_{hh}h_{t-1} + b_h) \tag{2}$$

$$y_t(t) = W_{yh}h_t + b_y \tag{3}$$

Where W_{hx}, W_{hh}, W_{yh} imply the weights of each step. It should be noted that what makes the RNN recurrent is to apply the same weights in each step. In particular, only three sets of weights are used by a typical vanilla RNN to make calculations. W_{hh} is employed for all $h_{t-1} \rightarrow h_t$ links, W_{xh} is applied for all $x_t \rightarrow h_t$ links, and W_{yh} is employed for all $h_t \rightarrow y_t$ links. Also, a couple of biases have been applied for RNN: b_h and b_y.

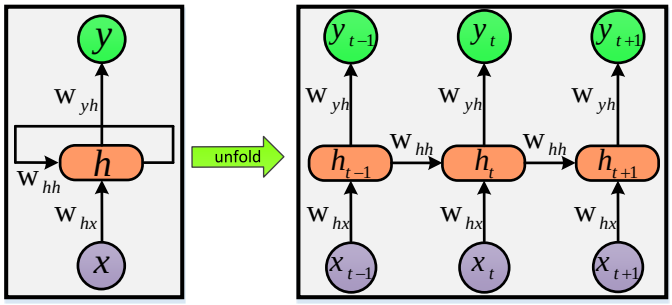

FIGURE 1 The structure of the simple RNN and unfolded RNN.

2.2 | GRU-RNN architecture

The RNN may be used as a network memory, different from the feedforward neural network (FNN). Consequently, the current state is associated with the previous state and the current input. That helps the RNN handle time series problems by storing,

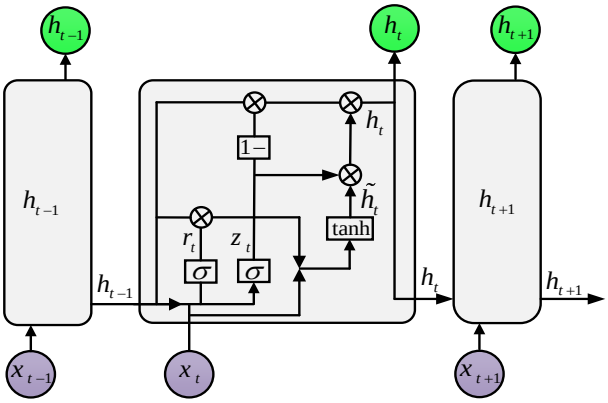

FIGURE 2 The structure of gated recurrent unit memory cell.

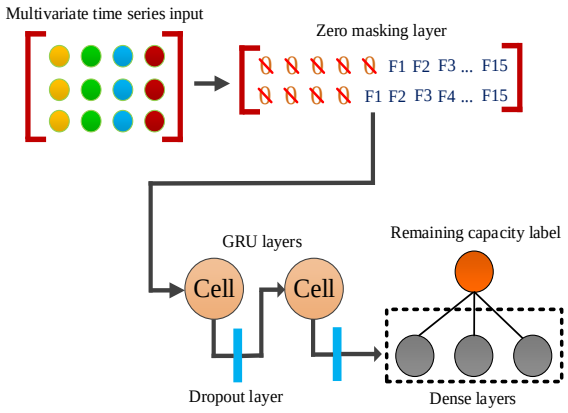

FIGURE 3 The structure of deep learning model for RUL prediction.

preserving, and evaluating the previous complex signals over a specific time. RNNs are extensively applied in numerous applications such as prediction of time series, system modeling, and natural language processing. Nevertheless, complex hidden layers and long time series may contribute to the exploding and vanishing of gradients throughout back-propagation procedures. Across all the enhanced RNNs, the GRU-RNN not only has a simple structure but also able to capture long-term sequential dependencies. Furthermore, the gradients are more resistant to vanishing compared to other RNNs, and fewer memory resources are needed. Therefore, GRU-RNN is ideal for dealing with highly correlated issues with time series, such as RUL prediction of the battery system. The structure of the GRU-RNN cell and its deep learning model are shown in Fig. 2 and Fig. 3 respectively, which will be explained in Section 3.2.

Behind the GRU-RNN idea, there are two main key parameters, which are called as update gate and reset gate. Both the reset gate r_t and update gate z_t are relevant to x_t and h_{t-1}. x_t is the corresponding input sequence, and h_{t-1} is the memory cell output at the previous time point. These two gates have distinct network functions. Four primary Equations (4) to (7) are used to calculate the GRU-RNN forward propagation.

- **Update gate**: The update gate is developed to monitor the previous data's effect on the current state. The bigger the updated value is, the more previous information is utilized to specify the current state. Eq. (4) reflects the GRU-RNN update operation.

$$z(t) = \sigma(w_z.[h_{t-1}, x_t] + b_z) \tag{4}$$

- **Reset gate**: The reset gate controls the level of ignorance of information in h_{t-1}. If the value of the reset gate is small, the information is more overlooked. This parameter can be applied for the prediction of RUL of lithium-ion batteries for

rejecting outliers, noises, and unnecessary degradation information between adjacent cycles[29]. This is because the structure of RNN-based model which considers long-term information, and this leads to suppress the effect of weight at the adjacent cycle data.[30]. Eq. (5) and Eq. (6) reflect the GRU-RNN reset operation.

$$r(t) = \sigma(w_r.[h_{t-1}, x_t] + b_r) \tag{5}$$

$$\tilde{h}(t) = \tanh(w_{\tilde{h}}.[r_t \odot h_{t-1}, x_t] + b_{\tilde{h}}) \tag{6}$$

- **Output**: Eq. (7) indicates the GRU-RNN output operation.

$$h(t) = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{7}$$

The derivatives of σ and tanh are the function of the primary function, therefor the derivatives can be computed by the primary functions.

$$\sigma(x) = \frac{1}{1 + \exp(-x)} \tag{8}$$

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)} \tag{9}$$

3 | GRU-RNN-ORIENTED RUL PREDICTION

In order to clarify the steps taken for both feature engineering and RUL prediction methods, the framework of the proposed method is indicated in Fig. 4.

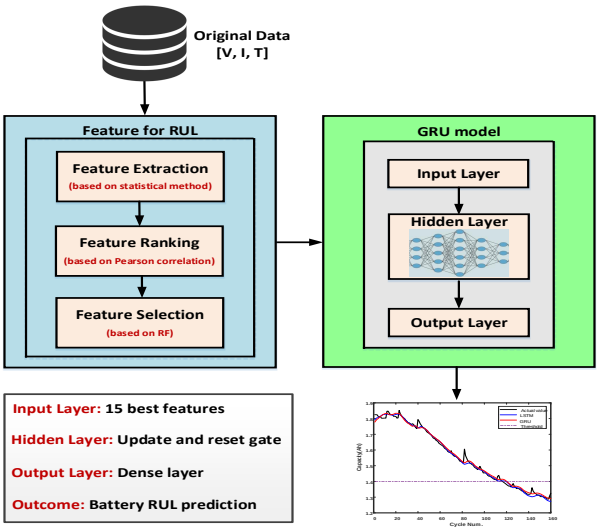

FIGURE 4 Framework of battery remaining useful life prediction.

3.1 | Feature engineering

Before training of the model, it is necessary to do pre-processing on raw data. The statistical characteristics are effective methods with the advantages of low time consumption for calculation and also simplicity of implementation[31]. Many studies have been

used statistical features for diagnosis and prognosis problems[32], and it has shown strong results. RMS, Kurtosis, Skewness, Peak-Peak, Mean are the most commonly used time-domain features in industrial applications[33].

The measured terminal voltage and load current of any battery will vary as it is charged and discharged. The essential characteristics such as the nominal voltage of the cell, peak charged, and end of life (EOL) can be extracted from each charge and discharge curve in cycles. In this work, battery characteristic during discharge mode has been used for feature extraction. The terminal and current voltage profiles of a charge-discharge life cycle for 4 battery cells (called B0005, B0006, B0007, B0018) are indicated in Fig. 5 which will be explained in more details in section 4.1.

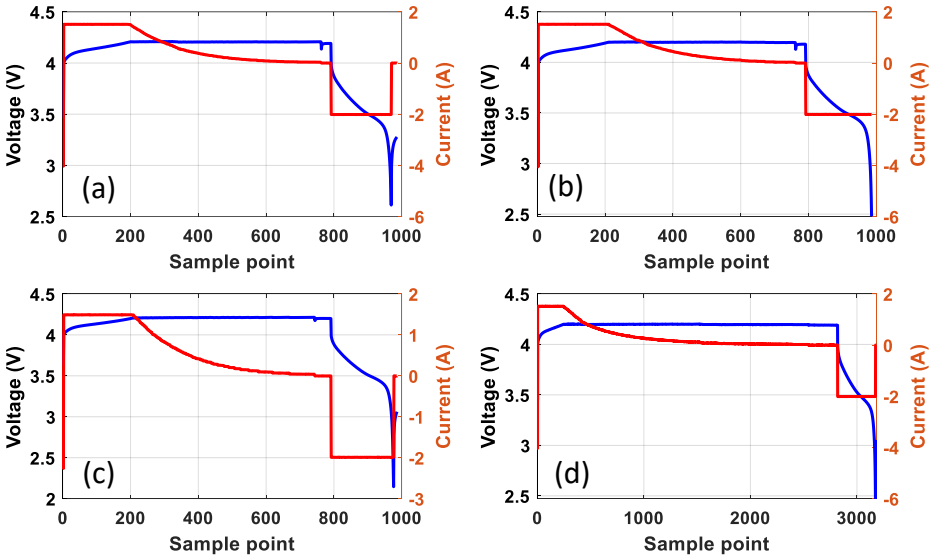

FIGURE 5 The terminal and current voltage profiles of a charge-discharge life cycle; (a) B0005, (b) B0006, (c) B0007, (d) B0018.

In our case, the feature engineering problem is divided into three parts:

1. 30 time-domain features are extracted from battery signals, including voltage, current, and temperature, based on statistical equations. These formulas are listed in Table 1.

TABLE 1 Statistical formulas in the time-domain.

Name	Formula	Name	Formula	Name	Formula
Mean	$F_m = \frac{1}{N}\sum_i^N x(i)$	Shape Factor (SHF)	$F_{shf} = \frac{F_{rms}}{\frac{1}{N}\sum_i^N \lvert x(i)\rvert}$	Skewness Factor (SF)	$F_{sf} = \frac{\frac{1}{N}\sum_i^N \lvert x(i)\rvert^3}{{F_{rms}}^3}$
Standard Deviation (STD)	$F_{std} = \sqrt{\frac{1}{N}\sum_i^N (x(i)-\overline{x})}$	Crest Factor (CF)	$F_{cf} = \frac{F_p}{F_{rms}}$	Kurtosis Factor (KF)	$F_{kf} = \frac{\frac{1}{N}\sum_i^N \lvert x(i)\rvert^4}{{F_{rms}}^4}$
Root Mean Square (RMS)	$F_{rms} = \sqrt{\frac{1}{N}\sum_i^N x(i)^2}$	Impulse Factor (IF)	$F_{if} = \frac{F_p}{\frac{1}{N}\sum_i^N \lvert x(i)\rvert}$	Clearance Factor (CF)	$F_{clf} = \frac{F_p}{\frac{1}{N}\sum_i^N \sqrt{\lvert x(i)\rvert}}$
Peak	$F_p = \max \lvert x(i)\rvert$	–	–	–	–

$x(i)$ is the battery signals series; $\overline{x}$ is the mean value of the series.

2. These features are sorted according to feature importance and separated the features based on the Pearson correlation, which has correlated with the coefficient above 0.5. For the purpose of quantitative confirmation of the linear correlation between the capacity and extracted features, Pearson correlation analysis can be used, which is computed as[33]:

$$r = \frac{\sum_{i=1}^{n}(HI_i - \bar{HI})(C_i - \bar{C})}{\sqrt{\sum_{i=1}^{n}(HI_i - \bar{HI})^2}\sqrt{\sum_{i=1}^{n}(C_i - \bar{C})^2}} \tag{10}$$

where HI, C, $\bar{HI}$, and $\bar{C}$ denote to feature, capacity, mean values of the HI, and mean values of the capacity, respectively. The value of the Pearson correlation coefficient r ranges between -1 and +1. If the correlation is equal to ± 1, features and capacity are completely correlated linearly, and no correlation exists in the case that it equals 0.

3. Finally, feature selection has been made based on the random forest algorithm owing to eliminating the less-relevant features and effective data training. To illustrate the importance of pre-processing, Fig. 6 shows that the average of the extracted features coefficient for pre-processing is 0.76, and after pre-processing is 0.88. To better understand the problem, the process of selecting the best features by RF algorithm is described below.

Random forest is an intelligent ensemble learning algorithm based on a decision tree, which contains a group of structured tree classifiers $h(x, \Theta_k), (k = 1, 2, 3, ...)$, in which a unit vote is cast by each tree for the especially known class at input x and Θ_k is identically distributed random vectors and also independent[34]. A margin function $mg(.)$, which is referred to as the confidence level for the RF model, needs to be defined with a performance index,

$$mg(x, y) = av_k I(h_k(x, \Theta_k) = y) - \max_{j \neq y} av_k I(h_k(x, \Theta_k) = j) \tag{11}$$

In this equation, $I(.)$ implies the indicator function as well as $av(.)$ is a mean value. This index is divided into two terms: the first term is referred to as the average number of votes at $(\mathbf{x}, \mathbf{y})$ for the right class, and the other term implies the average vote for the most class excluding the right class. In the case of large values of the margin, the confidence level will be high. Afterward, the generalization error PE^* is obtained through the following equation:

$$PE^* = P_{x,y}(mg(\mathbf{x}, \mathbf{y}) < 0) \tag{12}$$

Where $P(.)$ denotes the probability. By increasing the number of trees, the convergence of nearly all sequences Θ_k, PE^* is tend toward the following equation:

$$P_{x,y}(P_\Theta(h(\mathbf{x}, \Theta) = \mathbf{y}) - \max_{j \neq y} P_\Theta(h(x, \Theta) = j) < 0) \tag{13}$$

When the generalization error is converged, considerably low value for generalization error can be produced by RF. Also, RF does not overfit by adding more trees. The upper bound for the PE^* is determined by

$$PE^* \leq \frac{\rho(1 - s^2)}{s^2} \tag{14}$$

where ρ denotes the correlation average value, s implies the strength of each tree in the RF model. Hence, with decreasing the correlation among trees and raising the strength of each tree, the RF model would give higher precision of the predictions. For the purpose of showing the performance of the features selected by random forest, the heatmap is plotted which $F_1, F_2, ..., F_{15}$, are 15 selected features and has a high correlation with Cap as output, which is the battery capacity.

3.2 | GRU-RNN training

As it can be seen in Fig. 3, one input layer fed into a GRU layer with 50 neurons was applied to build the GRU-RNN. In turn, this layer is fed into two hidden layers with 50 neurons for both of them, which then fed into a fully connected dense layer of 20 neurons. Given that the battery data is time series, this work takes into account the time dependency, which has been overlooked in most existing papers, and it was not a delay (time lag) for training data. Due to the use of multivariate time series input, the zero masking layer has been considered for sequence processing. Moreover, the third layer is a fully connected dense

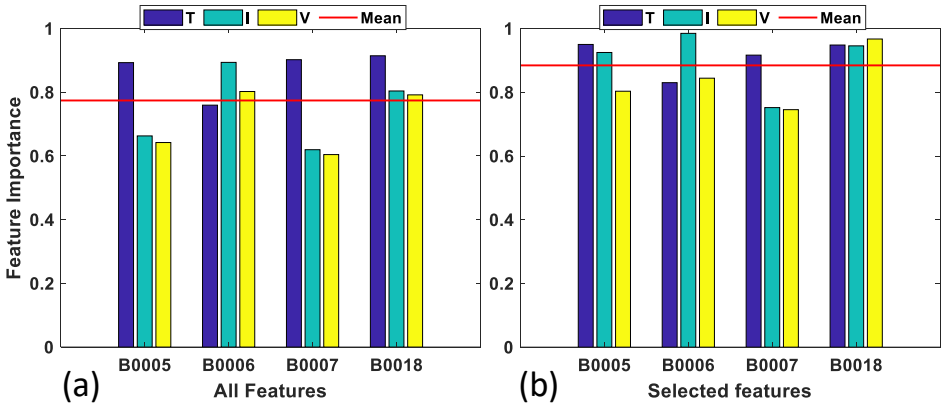

FIGURE 6 Feature extraction and selection; (a) shows all statistical features coefficient correlation; (b) shows selected features correlation coefficient.

layer applying a linear transformation for achieving the RUL prediction results using the sigmoid activation function, which is performed as follows:

$$RUL_t^* = \sigma(W_t.h_t + b_s) \tag{15}$$

where W_t and b_s respectively denote to the weight vector and biases of the fully connected layer at time step t. The mean absolute error (MAE) is chosen as the loss function, and it is calculated as indicated in Eq. 16:

$$L = (\sum_{t=1}^{l}(RUL_t - RUL_t^*)^2)/l \tag{16}$$

where RUL_t implies the measured value, RUL_t^* denotes to the predicted value, and l is the length of the battery discharge cycles.

3.3 | Adam optimization algorithm

Deep learning scientists have often sought to improve the model's efficiency and loss function value by the model's training epochs. Stochastic gradient descent (SGD) is one of these approaches that give a single learning rate for all weight updates and does not modify the learning rate throughout the training [35]. Nevertheless, this method is not effective for the training model due to frequent fluctuations; it will keep overshooting near to the desired exact minima and very time-consuming to converge to the correct network weights, which is inapplicable for online battery RUL prediction. Root mean square propagation (RMSprop) is another commonly used optimization method that overcomes the decaying learning rate problem of the SGD method. However, both of them still have the problem of different momentums for different parameters.

Therefore, the Adam algorithm was proposed by Diederik Kingma [36] to introduce the concept of adaptive momentum along with the adaptive learning rate, which computes the exponentially decaying average of previous gradients along with an adaptive learning rate. The key advantages of using the Adam algorithm in convex optimization issues are invariant to limited memory requirements and diagonal rescale of gradients. The Adam optimizer is a hybrid version of the AdaGrad, and RMSProp algorithms [37,38].

3.4 | Early stopping technique to prevent overfitting

Early stopping is widely used to implicitly regularize some convex learning problems [39]. Since the understanding and implementation are simple and have been reported to be superior to regularization methods in many studies, e.g., [40]. During the training, the model is evaluated on a holdout validation dataset after each epoch. The training process is stopped if the model's performance on the validation datasets starts to deteriorate (i.e., the loss is beginning to rise or accuracy is beginning to decrease). This technique is referred to as an early exit, so this is called an early stopping and is one of the most frequently used ways of regularizing neural networks. Its success lies in its quality and simplicity. In some papers, results confirm that early stopping could potentially improve generalization performance.

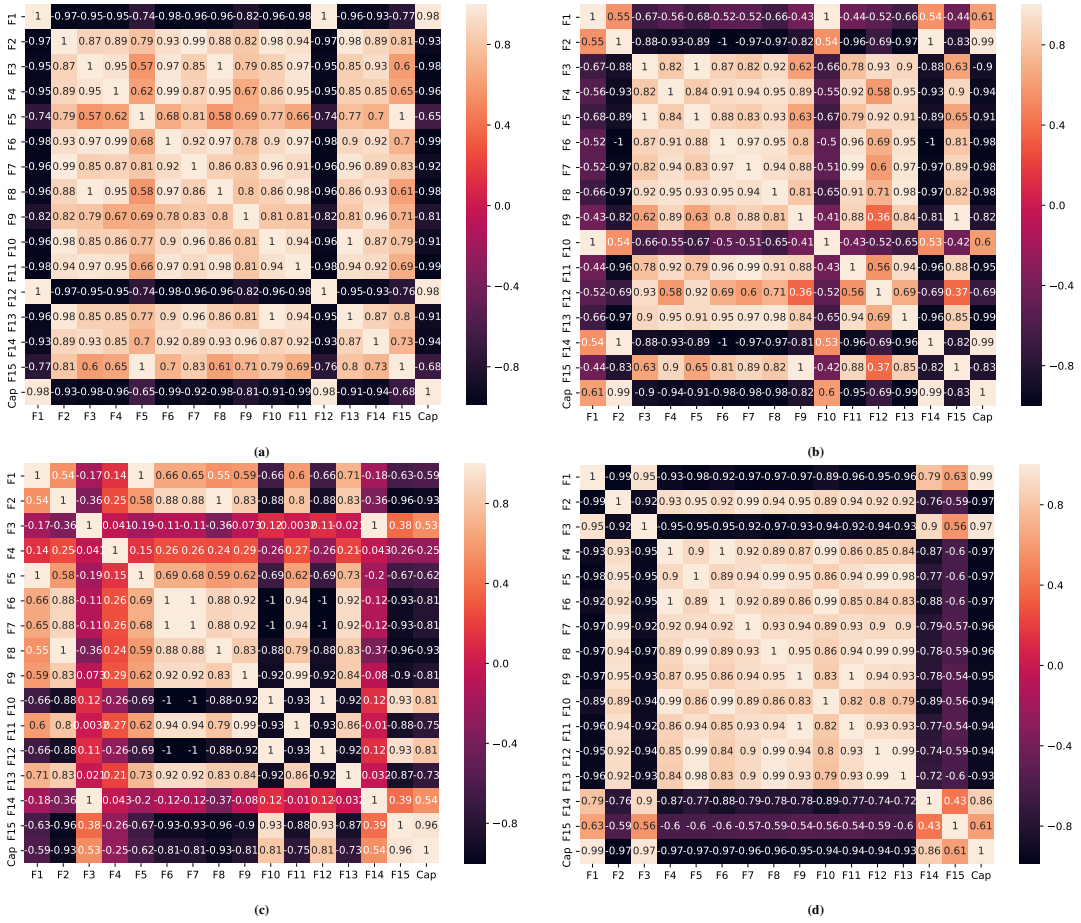

FIGURE 7 Heatmap of selected features using RF algorithm; (a) B0005, (b) B0006, (c) B0007, (d) B0018.

4 | RESULTS AND DISCUSSION

Matlab 2019 performed data preprocessing and feature engineering in this work, and Python 3.6 was also used for model simulation and training. The simulation was performed on a laptop with a graphic card NVIDIA GeForce 930M at 6 GB, 64-bit operating system and an Intel Core $i7-6500U$ processor (6 MB cache, up to 3.18 GHz), x64-based processor.

For a fair comparison, model parameters (such as learning rate=1e-5, lag=8, epoch=1000) are considered the same values. Besides, the batch size tuning has been done by a common method, which is called the grid search method. We set an early stopping for both models if the validation accuracy does not increase for 1000 global steps; the training will stop. It is worth noting that, due to the use of the *reduceLR* technique, the impact of the learning rate has diminished, and it has been no need for exact tuning.

4.1 | Data description and evaluation criteria

In the present article, the public battery data set of NASA Ames Prognostics Center of Excellence (PCoE) is used to validate our proposed method. Three different operational profiles (an impedance, discharge, charge) were used at the room temperature to run four Li-ion batteries (B0005, B0006, B0007, B0018). In this article, the authors only used the discharging mode for RUL prediction. Discharging was performed at a constant current (CC) level of 2A until the falling of the voltage of batteries B0005, B0006, and B0007 to 2.7V, 2.5V, and 2.2V, respectively. Repeated charge and discharge cycles lead to rapid battery aging, although impedance measurements give an overview of internal battery parameters altering with the progression of the aging process. The tests were completed in the case that the battery reached end-of-life (EOL) criteria on a 30% fade in rated capacity (from 2Ahr to 1.4Ahr)[41].

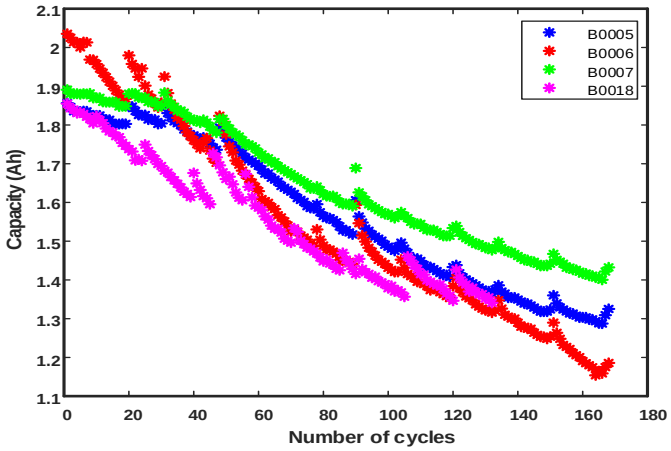

FIGURE 8 NASA Dataset.

The concept of building a validation set is to evaluate the model's performance prior to applying it to make predictions. The development of a validation set for time series issues is challenging since the time component must be considered. They represent calculation precision and are often applied to compare the pros and cons of algorithms. As the train-test-split or k-fold validation cannot be used directly, the pattern will be disrupted in the series. Hence, three evaluation criteria of RUL prediction error, RMSE, and MAE are used to calculate and demonstrate the suggested approach's precision and stability. The equations are given in Eq. (4-19).

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(\hat{Y}(i) - Y(i)\right)^2} \tag{17}$$

$$MAE = \frac{1}{m}\sum_{i=1}^{m}\left|\left(\hat{Y}(i) - Y(i)\right)\right| \tag{18}$$

$$RUL_{error} = RUL_{predict} - RUL_{true} \tag{19}$$

where $Y(i)$ and $\hat{Y}(i)$ denoted to the predicted capacity and measurement capacity series, respectively. i is the number of cycles between the actual battery and first prediction cycle. Besides, to assess the uncertainty quantification of the proposed model, the 95% confidence interval (CI) is performed for evaluation of the uncertainty as

$$95\%CI = \hat{\overline{Y}}(i) \pm 1.96 \times \sigma^2(\hat{Y}(i)) \tag{20}$$

where 95%CI is the confidence interval for RUL prediction. $\hat{\overline{Y}}(i)$ and σ^2 denote mean values of RUL prediction and variance of the predicted values, respectively.

4.2 | RUL prediction results for battery degradation data

Four datasets from Fig. 8 show an accelerating aging process obtained from the discharge mode of the battery. It shows that the battery degradation goes down during the time due to internal reactions in charging and discharging cycles. To verify the proposed model, we considered two scenarios, including training the model with 60% dataset and the other one with 80%. In the first scenario, Fig. 9 is shown the battery RUL prediction with 60% training data for four different cases in which the start point for prediction is from the 97^{th} cycle for B0005, B0006, B0007 and 75^{th} cycle for B0018. Fig. 11 is shown the prediction error for both GRU-RNN and LSTM-RNN as well. To indicate the GRU-RNN model's accuracy, a comparison has been made between the GRU-RNN, LSTM-RNN, and SVM methods. Fig. 10 is shown a box plot of all training and testing errors together. GRU-RNN has the lowest error for all the cases, and the LSTM-RNN has less accuracy than GRU-RNN. Meanwhile, the SVM has a big difference from them, which is not suitable for the long-term dependency prediction. Given that the evaluation of models with MAE, RMSE, and RUL prediction, Table. 2 provided a comparison between the models. For the second scenario,

we applied 80% data for training and compared the different methods. Fig. 12 is shown the battery RUL prediction with 80% training data for four different cases, which the start point for prediction is from the 129^{th} cycle for B0005, B0006, B0007 and 100^{th} cycle for B0018. The prediction error is shown in Fig. 14, and also the box plot of all training and testing error illustrates the GRU-RNN has less amplitude of error than the other methods. For better methods comparison, Table. 3 is provided the MAE, RMSE, and RUL prediction error. The results achieved in the above section show that the deep learning models are much more precise compared to a conventional method because of advantages such as capturing long-term dependency time series data.

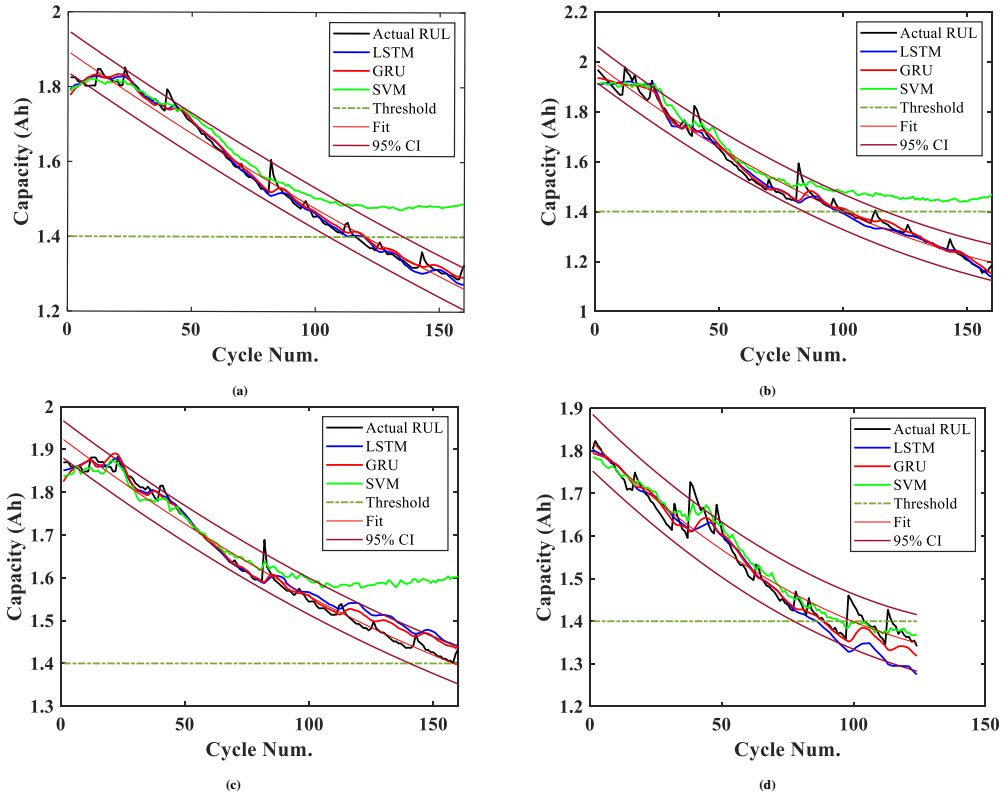

FIGURE 9 RUL prediction (60% of training); (a) B0005, (b) B0006, (c) B0007, (d) B0018.

In addition, to highlight the proposed method, the average RMSE percentage for all cases has been listed in Table 4. This Table shows the GRU-RNN error is about 2% which is more accurate compared to its peer and appropriate for real-world systems. Moreover, the executed time for GRU-RNN is about 14 seconds which demonstrates that due to fewer parameters, the learning speed is faster than the speed of LSTM (19 seconds). However, SVM has a high execution speed due to its straightforward structure, but it is not suitable for time series problems and has very weak prediction accuracy.

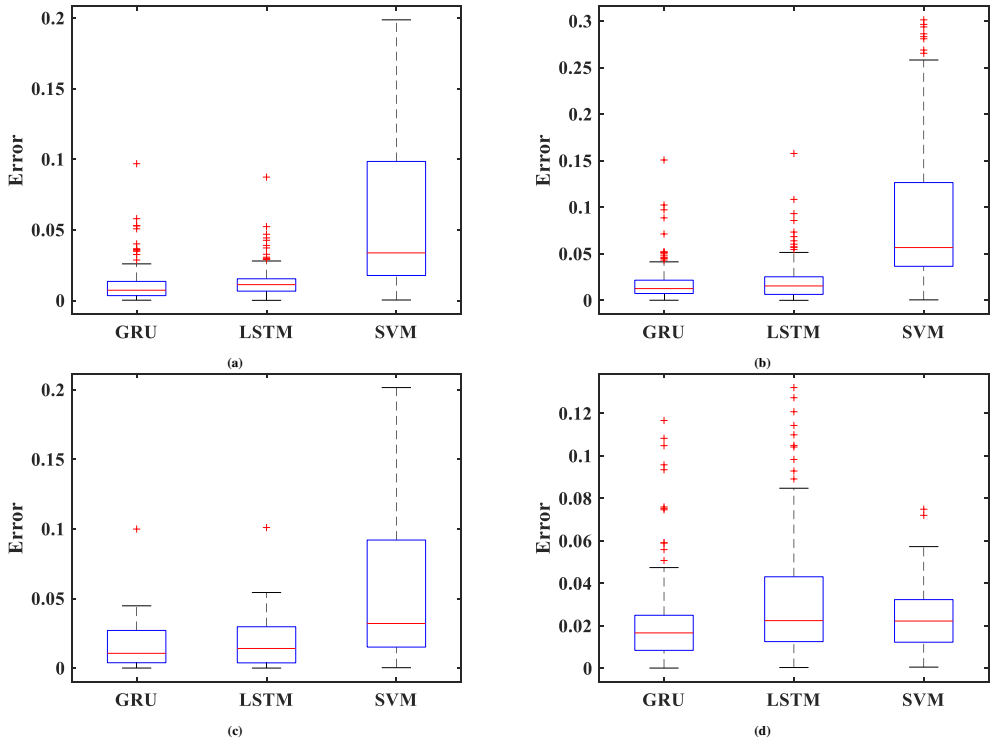

FIGURE 10 Box plot of prediction error (60% of training); (a) B0005, (b) B0006, (c) B0007, (d) B0018.

TABLE 2 RUL prediction results with 60% of training.

Battery cell	Method	MAE	RMSE	RUL error
B0005	GRU	0.0115	0.0145	0.6213
	LSTM	0.0124	0.0174	0.6337
	SVM	0.1328	0.1385	6.7772
B0006	GRU	0.0127	0.0165	0.6510
	LSTM	0.0136	0.0211	0.6878
	SVM	0.1777	0.1884	9.0664
B0007	GRU	0.0268	0.0290	1.3680
	LSTM	0.0367	0.0392	1.8753
	SVM	0.1283	0.1349	6.5481
B0018	GRU	0.0267	0.0389	1.0681
	LSTM	0.0561	0.0657	2.2460
	SVM	0.0302	0.0455	1.8116

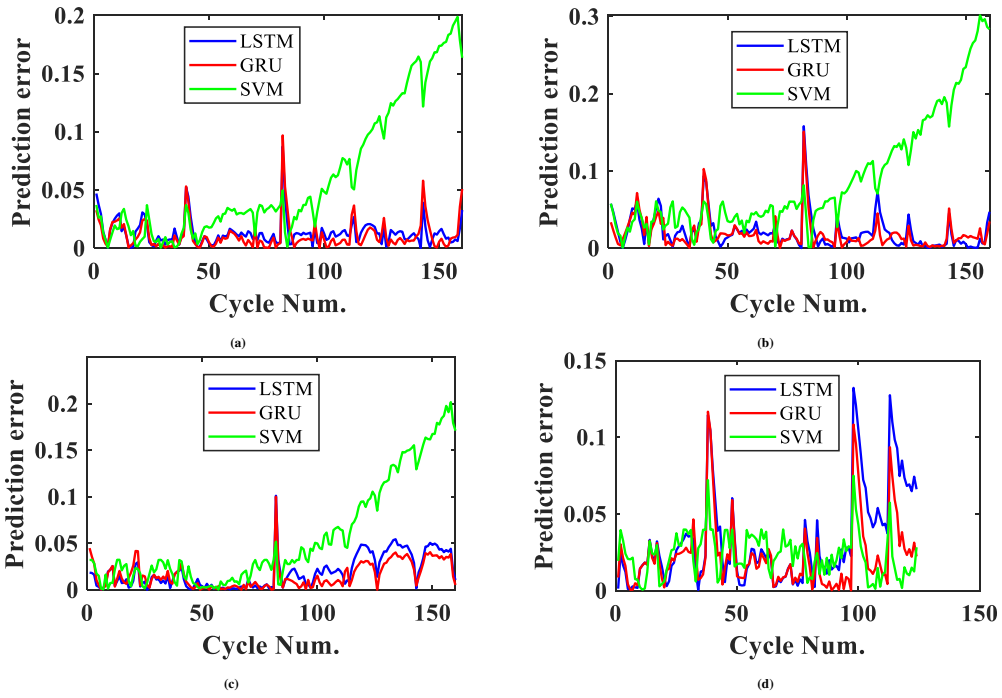

FIGURE 11 Prediction error (60% of training); (a) B0005, (b) B0006, (c) B0007, (d) B0018.

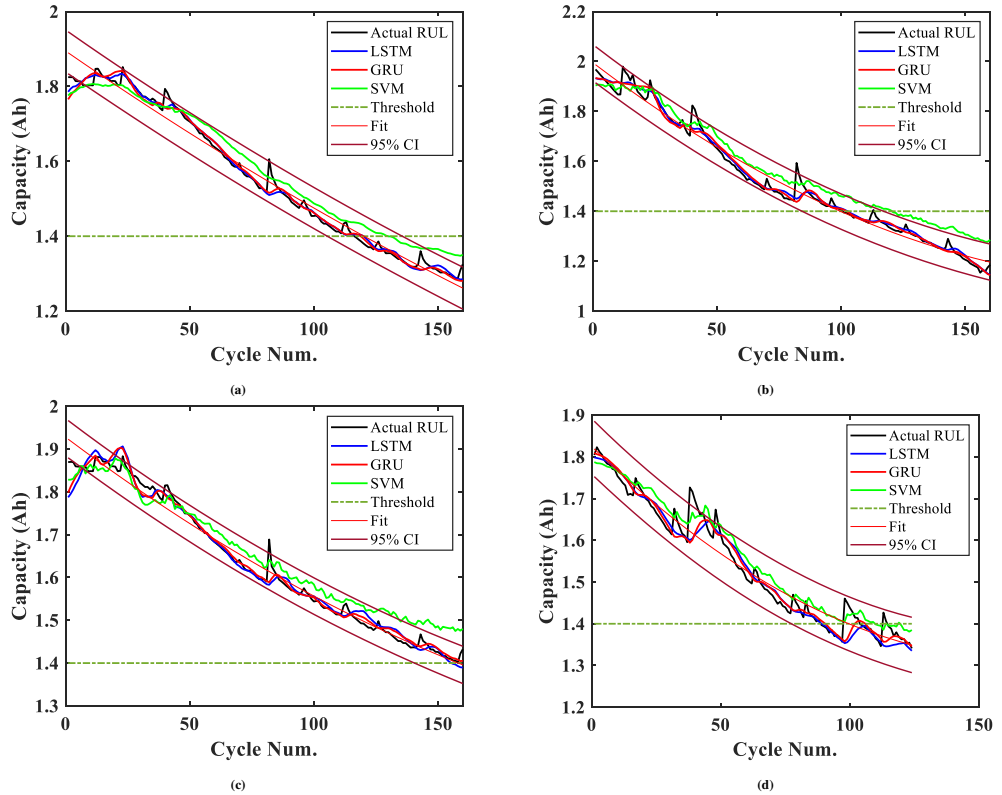

FIGURE 12 RUL prediction (80% of training); (a) B0005, (b) B0006, (c) B0007, (d) B0018.

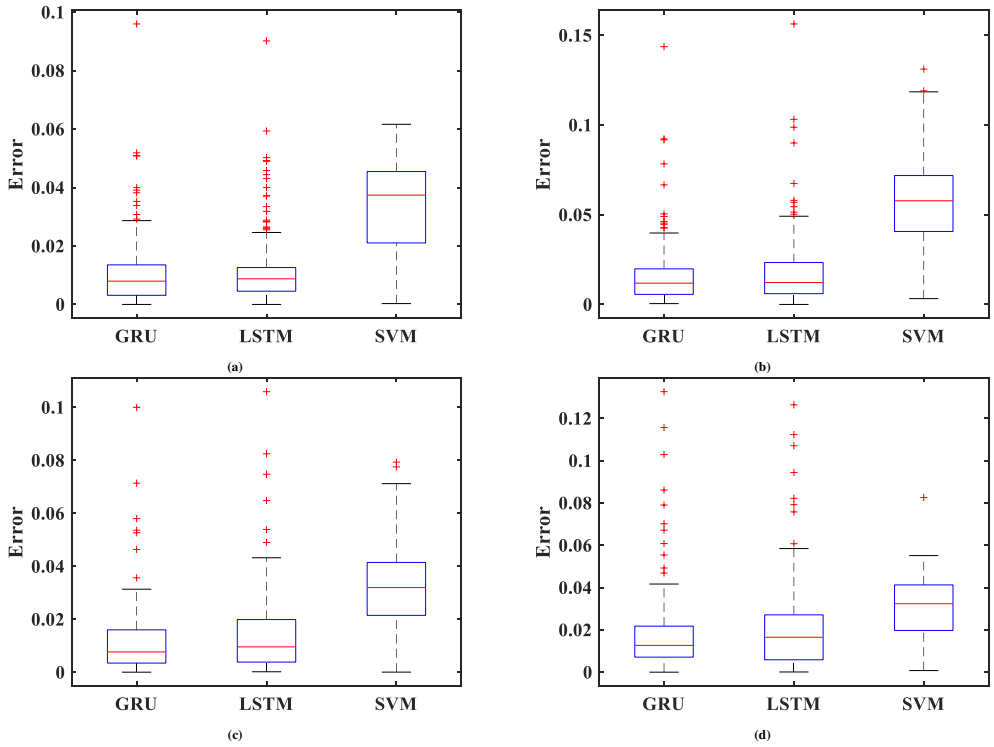

FIGURE 13 Box plot of prediction error (80% of training); (a) B0005, (b) B0006, (c) B0007, (d) B0018.

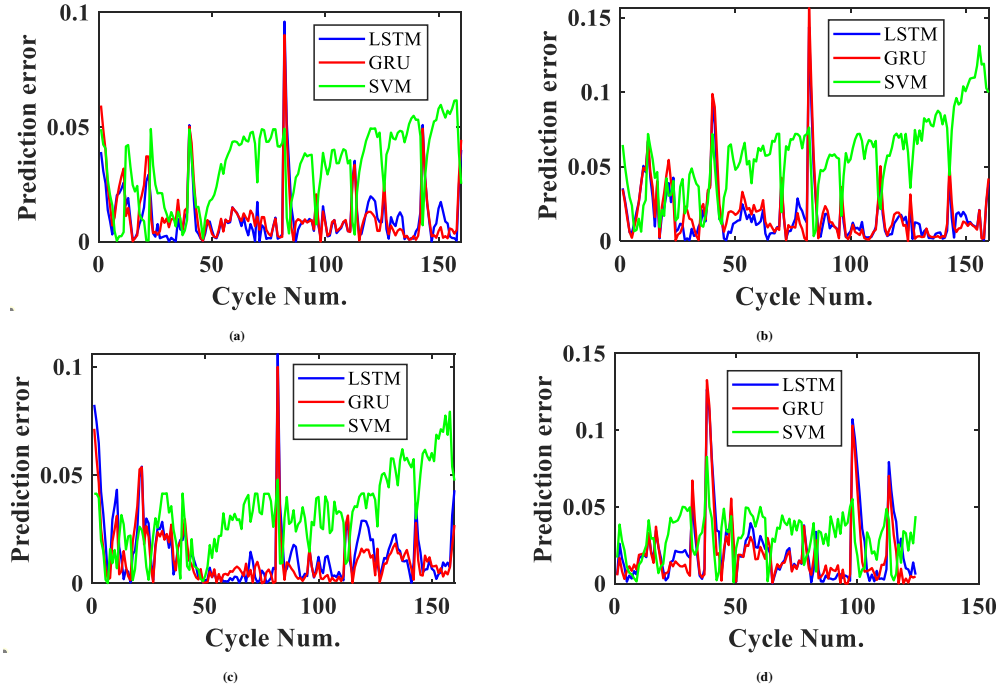

FIGURE 14 Prediction error (80% of training); (a) B0005, (b) B0006, (c) B0007, (d) B0018.

TABLE 3 RUL prediction results with 80% of training.

Battery cell	Method	MAE	RMSE	RUL error
B0005	GRU	0.0099	0.0156	0.1698
	LSTM	0.0117	0.0158	0.1990
	SVM	0.0500	0.5122	0.8504
B0006	GRU	0.0133	0.0165	0.2267
	LSTM	0.0147	0.0172	0.2504
	SVM	0.1042	0.1050	1.7744
B0007	GRU	0.0089	0.0106	0.1526
	LSTM	0.0115	0.0155	0.1958
	SVM	0.0615	0.0624	1.0461
B0018	GRU	0.0163	0.0259	0.2295
	LSTM	0.0249	0.0337	0.3492
	SVM	0.0265	0.3025	0.3722

TABLE 4 The average of RMSE prediction accuracy and executed time for all cases.

	SVM	LSTM	GRU
RMSE (%)	17.43	2.82	2.09
Executed time (s)	0.9511	19.2397	14.1173

5 | CONCLUSION

As a crucial tool for prognostic and health management (PHM), the remaining useful life (RUL) prediction is capable of ensuring a possible Li-ion battery failure time in advance. One of the most crucial concerns in the RUL prediction of Li-ion batteries is the way of appropriately learning the long-term dependencies of several hundred cycles while limited degradation data are available.

This paper has been presented for a data-driven model to monitor battery health. The gated recurrent unit (GRU) recurrent neural network (RNN) has been used to predict the battery RUL. To achieve high accuracy prediction, important features based on Pearson correlation and random forest (RF) algorithm have been applied to feed into the GRU-RNN as a multivariate input. Moreover, to optimize the training network, the Adam technique has been applied for convex optimization, which requires low memory. At the same time, an early stopping technique has been used to deal with overfitting and leads to enhance the performance of the GRU-RNN model.

For the experimental and evaluation of our proposed method, the NASA Li-ion battery data set has been applied. The findings have been compared with its sibling technique, which is called LSTM. The results highlight the proposed method has higher accuracy and efficiency than LSTM-RNN and SVM.

References

1. Liu Kailong, Li Kang, Peng Qiao, Zhang Cheng. A brief review on key technologies in the battery management system of electric vehicles. *Frontiers of Mechanical Engineering.* 2019;14(1):47–64.
2. Liu Zhi, Mohammadzadeh Ardashir, Turabieh Hamza, Mafarja Majdi, Band Shahab S, Mosavi Amir. A New Online Learned Interval Type-3 Fuzzy Control System for Solar Energy Management Systems. *IEEE Access.* 2021;9:10498–10508.
3. Naha Arunava, Khandelwal Ashish, Hariharan Krishnan S, Kaushik Anshul, Yadu Ankit, Kolake Subramanya Mayya. On-board short circuit detection of Li-ion batteries undergoing fixed charging profile as in smartphone applications. *IEEE Trans. Ind. Electron.*. 2019;.
4. El Mejdoubi Asmae, Chaoui Hicham, Gualous Hamid, Van Den Bossche Peter, Omar Noshin, Van Mierlo Joeri. Lithium-ion batteries health prognosis considering aging conditions. *IEEE Trans. Power Electron.*. 2018;34(7):6834–6844.
5. Liao Linxia, Köttig Felix. Review of hybrid prognostics approaches for remaining useful life prediction of engineered systems, and an application to battery life prediction. *IEEE Trans. Rel.*. 2014;63(1):191–207.
6. Shen Dongxu, Wu Lifeng, Kang Guoqing, Guan Yong, Peng Zhen. A novel online method for predicting the remaining useful life of lithium-ion batteries considering random variable discharge current. *Energy.* 2021;218:119490.
7. Chen Lin, Wang Huimin, Chen Jing, et al. A novel remaining useful life prediction framework for lithium-ion battery using grey model and particle filtering. *International Journal of Energy Research.* 2020;44(9):7435–7449.
8. Chang Yang, Fang Huajing, Zhang Yong. A new hybrid method for the prediction of the remaining useful life of a lithium-ion battery. *Applied energy.* 2017;206:1564–1578.
9. Lyu Zhiqiang, Gao Renjing. Li-ion battery state of health estimation through Gaussian process regression with Thevenin model. *International Journal of Energy Research.* ;.
10. Shamshirband Shahab, Fathi Mahdis, Chronopoulos Anthony T, Montieri Antonio, Palumbo Fabio, Pescapè Antonio. Computational intelligence intrusion detection techniques in mobile cloud computing environments: Review, taxonomy, and open research issues. *Journal of Information Security and Applications.* 2020;55:102582.
11. Shamshirband Shahab, Rabczuk Timon, Chau Kwok-Wing. A survey of deep learning techniques: application in wind and solar energy resources. *IEEE Access.* 2019;7:164650–164666.
12. Liu Kailong, Hu Xiaosong, Zhou Huiyu, Tong Lei, Widanalage Dhammika, Macro James. Feature Analyses and Modelling of Lithium-ion Batteries Manufacturing based on Random Forest Classification. *IEEE/ASME Transactions on Mechatronics.* 2021;.

13. Qummar Sehrish, Khan Fiaz Gul, Shah Sajid, et al. A deep learning ensemble approach for diabetic retinopathy detection. *IEEE Access.* 2019;7:150530–150539.

14. Shamshirband Shahab, Fathi Mahdis, Dehzangi Abdollah, Chronopoulos Anthony Theodore, Alinejad-Rokny Hamid. A Review on Deep Learning Approaches in Healthcare Systems: Taxonomies, Challenges, and Open Issues. *Journal of Biomedical Informatics.* 2020;:103627.

15. Hu Xiaosong, Che Yunhong, Lin Xianke, Onori Simona. Battery health prediction using fusion-based feature selection and machine learning. *IEEE Transactions on Transportation Electrification.* 2020;.

16. Ardeshiri Reza Rouhi, Balagopal Bharat, Alsabbagh Amro, Ma Chengbin, Chow Mo-Yuen. Machine Learning Approaches in Battery Management Systems: State of the Art: Remaining useful life and fault detection. In: :61–66IEEE; 2020.

17. Mahmoudi Mohammad Reza, Baleanu Dumitru, Qasem Sultan Noman, Mosavi Amirhosein, Band Shahab S. Fuzzy clustering to classify several time series models with fractional Brownian motion errors. *Alexandria Engineering Journal.* 2021;60(1):1137–1145.

18. Li Yihuan, Li Kang, Liu Xuan, Wang Yanxia, Zhang Li. Lithium-ion battery capacity estimationâĂŤA pruned convolutional neural network approach assisted with transfer learning. *Applied Energy.* 2021;285:116410.

19. Wang Zhi-Hao, Hendrick , Horng Gwo-Jiun, Wu Hsin-Te, Jong Gwo-Jia. A prediction method for voltage and lifetime of lead–acid battery by using machine learning. *Energy Exploration & Exploitation.* 2020;38(1):310–329.

20. Liu Kailong, Ashwin TR, Hu Xiaosong, Lucu Mattin, Widanage W Dhammika. An evaluation study of different modelling techniques for calendar ageing prediction of lithium-ion batteries. *Renewable and Sustainable Energy Reviews.* 2020;131:110017.

21. Liu Kailong, Li Yi, Hu Xiaosong, Lucu Mattin, Widanage Widanalage Dhammika. Gaussian process regression with automatic relevance determination kernel for calendar aging prediction of lithium-ion batteries. *IEEE Transactions on Industrial Informatics.* 2019;16(6):3767–3777.

22. Li Xiaoyu, Wang Zhenpo, Yan Jinying. Prognostic health condition for lithium battery using the partial incremental capacity and Gaussian process regression. *Journal of power sources.* 2019;421:56–67.

23. Liu Kailong, Hu Xiaosong, Wei Zhongbao, Li Yi, Jiang Yan. Modified Gaussian process regression models for cyclic capacity prediction of lithium-ion batteries. *IEEE Transactions on Transportation Electrification.* 2019;5(4):1225–1236.

24. Li Xiaoyu, Yuan Changgui, Li Xiaohui, Wang Zhenpo. State of health estimation for Li-Ion battery using incremental capacity analysis and Gaussian process regression. *Energy.* 2020;190:116467.

25. Li Weihan, Sengupta Neil, Dechent Philipp, Howey David, Annaswamy Anuradha, Sauer Dirk Uwe. Online capacity estimation of lithium-ion batteries with deep long short-term memory networks. *Journal of Power Sources.* 2021;482:228863.

26. Song Yuchen, Li Lyu, Peng Yu, Liu Datong. Lithium-Ion Battery Remaining Useful Life Prediction Based on GRU-RNN. In: :317–322IEEE; 2018.

27. Ungurean Lucian, Micea Mihai V, Cârstoiu Gabriel. Online state of health prediction method for lithium-ion batteries, based on gated recurrent unit neural networks. *International Journal of Energy Research.* 2020;44(8):6767–6777.

28. Venugopal Prakash, others . State-of-Health Estimation of Li-ion Batteries in Electric Vehicle Using IndRNN under Variable Load Condition. *Energies.* 2019;12(22):4338.

29. Zhang Yongzhi, Xiong Rui, He Hongwen, Pecht Michael G. Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries. *IEEE Trans. Veh. Technol..* 2018;67(7):5695–5705.

30. Choi Yohwan, Ryu Seunghyoung, Park Kyungnam, Kim Hongseok. Machine learning-based lithium-ion battery capacity estimation exploiting multi-channel charging profiles. *IEEE Access.* 2019;7:75143–75152.

31. Feng Jing, Kvam Paul, Tang Yanzhen. Remaining useful lifetime prediction based on the damage-marker bivariate degradation model: A case study on lithium-ion batteries used in electric vehicles. *Engineering Failure Analysis.* 2016;70:323–342.

32. Chen Cong, Xu Tianhua, Wang Guang, Li Bo. Railway turnout system RUL prediction based on feature fusion and genetic programming. *Measurement.* 2020;151:107162.

33. Mao Wentao, He Jianliang, Zuo Ming J. Predicting remaining useful life of rolling bearings based on deep feature representation and transfer learning. *IEEE Trans. Instrum. Meas.*. 2019;.

34. Zhang Dahai, Qian Liyang, Mao Baijin, Huang Can, Huang Bin, Si Yulin. A data-driven design for fault detection of wind turbines using random forests and XGboost. *IEEE Access.* 2018;6:21020–21031.

35. Ecer Fatih, Ardabili Sina, Band Shahab S, Mosavi Amir. Training Multilayer Perceptron with Genetic Algorithms and Particle Swarm Optimization for Modeling Stock Price Index Prediction. *Entropy.* 2020;22(11):1239.

36. Kingma Diederik P, Ba Jimmy. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980.* 2014;.

37. Chin Wei-Sheng, Zhuang Yong, Juan Yu-Chin, Lin Chih-Jen. A learning-rate schedule for stochastic gradient methods to matrix factorization. In: :442–455Springer; 2015.

38. Levy Omer, Goldberg Yoav. Neural word embedding as implicit matrix factorization. In: :2177–2185; 2014.

39. Zhang Chiyuan, Bengio Samy, Hardt Moritz, Recht Benjamin, Vinyals Oriol. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530.* 2016;.

40. Finnoff William, Hergert Ferdinand, Zimmermann Hans Georg. Improving model selection by nonconvergent methods. *Neural Networks.* 1993;6(6):771–783.

41. Fan Jiaming, Fan Jianping, Liu Feng, Qu Jiantao, Li Ruofeng. A Novel Machine Learning Method Based Approach for Li-Ion Battery Prognostic and Health Management. *IEEE Access.* 2019;7:160043–160061.