Signal Processing and Renewable Energy
March 2023, (pp. 1-12)
ISSN: 2588-7327
eISSN: 2588-7335

Remaining Useful Life Estimation Enhancement via Deep Adaptive Feature Extraction

Zahra Esfahani[1], Karim Salahshoor[2*], and Amir Hooshang Mazinan[3]

[1] Ph.D. Student, Elec. Eng., Islamic Azad Univ., South Tehran branch, Tehran, Iran.
[2] Prof., Dept. of Instrumentation and Automation., Petroleum Univ. of Tech., Ahwaz, Iran.
[3] Assoc. Prof., Islamic Azad Univ., South Tehran branch, Tehran, Iran.

Received: 14-Sep-2022, Revised: 24-Oct-2022, Accepted: 30-Oct-2022.

Abstract
The length between the current point in the degradation process and the time of reaching the failure threshold, or the remaining usable life (RUL) prediction of systems, is of the greatest priority in the industry. More accurate estimation is useful for maintenance decisions as it helps to avoid catastrophic breakdowns and may also assist in reducing additional costs. Deep learning approaches have made impressive advancements in this field in recent years by becoming widely attractive and employed. However, most deep learning approaches don't fully consider the information implications of sensors adaptively. To overcome this problem, a novel adaptive hybrid model that combines a convolutional neural network (CNN) and gated recurrent unit (GRU) is introduced in this work. The RUL estimation is based on the best practical option of sequence data through CNN-GRU. In the first step, optimal sensor selection is applied to the dataset to collect the most useful sensors. Then, the input data is transformed into a predefined range of values using standard and min-max scalars; in the next step, the normalized data is fed into the CNN-GRU model with an adaptive activation function for deep feature extraction, training, and RUL prediction. Utilizing CNN to extract features from the multivariate input data automatically, the features are then fed into the GRU layer to train the model for RUL prediction. To test the effectiveness of this framework, the suggested methodology is applied to the NASA Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) dataset. The findings demonstrate that CNN-GRU is capable of accurate RUL prediction. In addition, CNN-GRU outperforms CNN-LSTM and CNN-RNN in terms of computation efficiency and accuracy.

Keywords: Optimal sensor selection, Adaptive CNN-GRU, RUL, Turbofan Engine.

***Corresponding Authors Email:** salahshoor@put.ac.ir

1. INTRODUCTION

Condition-based maintenance is an intelligent prognostic technique that has a great deal of promise to boost performance and dependability in various industries. Condition monitoring is a key component of prognostics and health management (PHM) systems, and this type of maintenance enables more accurate prediction. Model-based, data-based, and hybrid techniques are the three main prognostic strategies; recent articles have focused more on the integration of these methodologies. RUL prediction can benefit from data-driven techniques due to their ability. One of the main challenges for data-driven techniques is developing a practical method for figuring out the optimal relationship between a system's or component's monitoring data and its RUL. On the other hand, more significant historical data is more helpful, and understanding how to interpret sensor information is useful for estimation.

In recent years, data-driven methodologies have increasingly included artificial intelligence, particularly deep learning techniques, to automate feature design and perform nonlinear transformations in its layers. Regression is performed in [1] using the multi-layer perceptron (MLP) and radial basis function (RBF), and a fusion approach based on the Kalman filter is developed to predict the RUL of turbofan engines. On the other hand, [2] uses a semi-supervised approach in which data with two labels and zero labels is supplied into the regression training dataset in order to learn features and predict the RUL. To estimate the turbofan RUL, authors in [3] used a feed-forward neural network model based on Bayesian regularization methods. This solution eliminates the requirement to normalize the entire dataset, which has an impact on model performance. The degradation in turbofan performance is a time series with a constant trend. Recurrent neural networks (RNNs) are better suited for prognostics of turbofan engines due to their structure and have produced outstanding results due to their capability for handling time series data and RUL estimation. In [4], RNN was used to build bearing health indicators for RUL prediction, and the results are acceptable. The authors of [5] proposed a method that uses PCA for dimension reduction after combining several filters. Methods like MLP and random forest are used to learn the fundamental model. Even though RNN achieves great performance, the fundamental drawback of this strategy is its inability to link two sets of similar, though separated, data and the vanishing gradient through learning. Hence, to learn models, a large training dataset is also required. Long short-term memory (LSTM) has internal loops that keep important information to address the vanishing gradients issue with RNN models [6].

On the other hand, CNN as the most common network in both supervised and unsupervised learning is used for the first time in [7] to predict RUL. Both the outcomes of feature learning and RUL predictions are improved by the use of supervised feedback. It has been reported that the temporal CNN, which is comprised of a unique 1-D convolution operation, has the potential to be effective for solving problems involving the prediction of time series [8]. Typically, CNN creates features by combining input data with various filters, and the most significant features are subsequently retrieved by pooling layers [9]. The necessity to record sensor measurements over an extended period of time is a drawback of health-index-based techniques, and [10] employed data from sequential time sampling at any time interval to address this issue. Furthermore, to increase network capabilities for RUL prediction, kernels are automatically chosen based on a new kernel factor.

The main advantage of hybrid models is that they combine various architectures to increase efficiency while considering the advantages of each architecture.

Then, LSTM-RNN is suggested as the solution to the gradient explosion and gradient disappearance issues. [11] used LSTM-RNN to predict battery energy status and RUL. [12] proposes an LSTM-CNN model with a time window to predict RUL. However, it is important to note that the LSTM's complex architecture consists of three gates—the forget gate, the input gate, and the output. While providing modeling accuracy, the complex structure also increases computational complexity. To make the LSTM-CNN structure simpler while maintaining its modeling capabilities, the gated recurrent unit (GRU) has been proposed [13].

To establish the turbofan engine RUL prediction, this research introduces an adaptive CNN-GRU network. The features that are effective on the degradation trend are extracted from relevant sensors' historical data, and the model is learned using the CNN-GRU method. And using the model, the RUL for motors is determined. The C-MAPSS dataset is used to perform the experiment. The results demonstrated that the CNN-GRU algorithm may be used for RUL prediction for turbofan engines with high accuracy and considerable robustness. Listed below are the scientific contributions made by this work:

First, using the high-dimensional data, sensors are selected for the network based on factors including their trendability, monotonicity, and prognosability; next, the technique of normalization is done to the chosen sensors in order to get them ready to be used in the network.

1. An ensemble network including 1-D CNN and GRU is introduced to extract the features, training, and RUL prediction.
2. Mish function is utilized as an activation function of the network to adaptive extraction and training for enhancing the accuracy of the model.
3. The outcome from the proposed prognostics approach in RUL prediction related evaluation metrics can be considered acceptable.

The remainder of this paper will be structured as described in the following sections. The proposed methodology of CNN-GRU and the details of RUL prediction based on the network is described in Section 2. Section 3, discusses the experimental results. The whole article is summarized in Section 4.

2. THEORETICAL BACKGROUND OF STUDY

2.1. Convolutional Neural Network (CNN)

A feed-forward neural network with a particular architecture known as CNN can extract useful information from input. Convolution operation and pooling layers are of the two main kinds of modules that comprise this network [14]. Using a kernel w, also known as the weight, CNN's convolution operation produces the output by moving it through the input data. Convolutional layers' ability to share similar weights to avoid the overfitting problem is their key benefit. In addition, pooling operations reduce computation costs and input dimensions by replacing an outline of nearby output for the network's output.

The i-th factor in this process will be the following with regard to x being the input and w being the kernel, both of which have n dimensions:

$$Y(i) = \sum_{i,j} x(i-j)w(j) \tag{1}$$

For j from 0 to $k-1$. Fig1. is a general illustration of the CNN path's convolution process.

2.2. Gated Recurrent Unit (GRU)

The Gated Recurrent Unit is the most recent technique in sequence modeling techniques, that also gives it a distinct advantage over RNN and LSTM [16].

The fundamental RNN structure consists of a cell with a cyclic loop whose internal state changes over time depending on the sample input now being used x_t and its previously hidden state h_{t-1} at each time step t. The current hidden state h_t is then updated as follows:

$$h_t = H(x_t, h_{t-1}) \tag{2}$$

H is a nonlinear, differentiable transform. Backpropagation trains RNN parameters. The recurrent network becomes deep as sequence length increases, causing decreasing gradient. Long short-term memory (LSTM) is one of the most used RNN models for vanishing gradient. LSTM uses a well-designed memory cell with an input gate, forget gate, and output gate to preserve and update the cell state.

GRU is a simplified variation of LSTM with two gates: a reset gate c that adjusts new input with previous memory and an update gate u that retains previous memory. Fig. 2 shows GRU cell architecture. GRU's fewer variables make it more efficient than LSTM. Here are GRU's hidden unit transition functions:

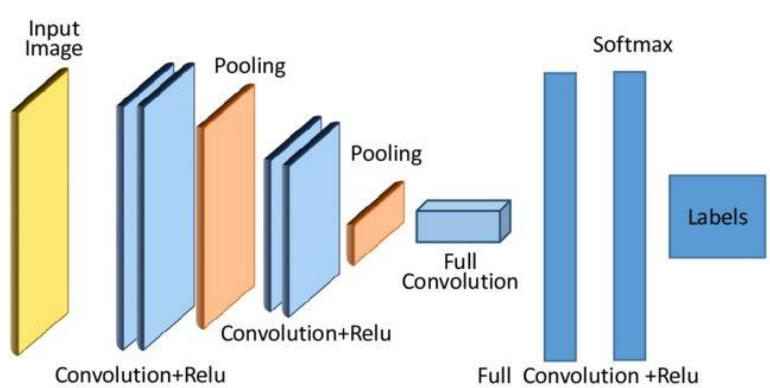

Fig. 1. A generic representation of CNN[15].

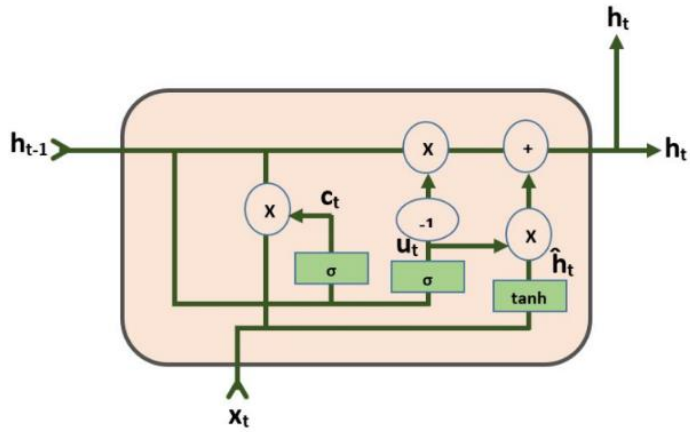

Fig. 2. The GRU architecture.

$$u_t = \sigma(F^u x_t + G^u h_{t-1} + b^u) \quad (3)$$

$$c_t = \sigma(F^c x_t + G^c h_{t-1} + b^c) \quad (4)$$

c_t is combined with a $tanh$ layer in order to get the new remember h_t,

$$\hat{h}_t = \tanh\,(F^c x_t + G^c(r_t \Theta h_{t-1})) \quad (5)$$

The value of the hidden state is continually updated by

$$h_t = (1 - u_t) - h_{t-1} + z_t - \hat{h}_t \quad (6)$$

wherein the model parameters, such as F, G, and b, are shared throughout all time steps, were learned during training, and indicate the element-wise product.

2.3. CNN-GRU for RUL Prediction

The normalization technique is used to scale the characteristic units of the data from the original historical sensor measurements. Additionally, it will be simple to understand the extracted features from the following phase. Second, the characteristics are obtained using CNN. Finally, the GRU is used to predict the model using the extracted features. To verify that there is enough information left in the recurrent network, the CNN-hidden GRU's layer is set to have a value of 4. It is beneficial to ensure that the suggested network is capable of high modeling and prediction accuracy.

3. EXPERIMENTAL RESULTS

3.1. Case Study

In this section, the proposed methodology is applied to a simulated turbofan engine dataset which is provided by NASA Prognostics Data Repository, and it is known as C-MAPSS Dataset [17]. This popular dataset consists of four different datasets as given in Table1 that these datasets contain simulated run-to-failure paths of turbofan engine with various operational conditions and fault modes. Furthermore, there are one training dataset and one testing dataset in each sub-dataset. 21 sensors which are indicated in Table2 are utilized to monitor and record run-to-failure data. Each dataset comprises a matrix with m-by-26 dimension in which m correlates with the number of data points of each turbofan engine. Rows indicate data collected during time cycles and columns illustrate engine number, operational cycle number, three different operational settings, and 21 sensor values respectively.

The main difference between the training dataset and the testing one is that the last data in the training dataset corresponds to the failure time of the engine but in the testing dataset records of sensors have stopped some time before failure and the estimation of the remaining useful life of the engine is provided with the testing dataset. On the other hand, actual values of RUL are provided to certify predicted RUL.

3.2 Data Pre-processing and Image Creation

According to the literature, the three main factors for selecting relevant sensors among multi-sensor measures are monotonicity, trend-ability, and prognosability. Using these three characteristics, useful sensors for the proposed network are chosen [18].

On the other hand, due to the large difference between maximum and minimum quantity of the turbofan engines data, it is compulsory to normalize data before pre-processing procedure. Moreover, the data cleaning process is necessary to eliminate noisy data and outliers in the dataset. Min-max and Z-score normalization methods are two beneficial ways to normalize data. Due to difficulties in the analysis of data with zero rows, min-max normalization is utilized in this paper and its formula is provided as follows.

$$x' = 2\frac{x-x_{min}}{x_{max}-x_{min}} - 1 \tag{7}$$

3.3 Evaluation Metrics

Because performance evaluation is critical for analyzing predicted models, this section includes various performance indicators for evaluating prognostics algorithms. These measurements can be divided into three categories: accuracy, precision, and robustness. On the other hand, there are numerous metrics that have been used in prognostics. As a result, this study gives a universal metric that is particularly useful for assessing error as the difference between the actual output and the aim. In this study, the Root Mean Squared Error (RMSE) is utilized, as stated below.

RMSE reveals the absolute measure of model fitting, and it can be described as the standard deviation of the unidentified variables. The formula for calculating RMSE is:

Table 1. ***Information on available datasets***.

	Datasets	# Fault Modes	# Operational Conditions	# Train Units	# Test Units
C-MAPSS	#1	1	1	100	100
	#2	1	6	260	259
	#3	2	1	100	100
	#4	2	6	249	248

Table 2. ***Details of 21 sensors for turbofan engine.***

Sensor No.	Symbol	Description
1	T2	Total temperature at fan inlet
2	T24	Total temperature at LPC outlet
3	T30	Total temperature at HPC outlet
4	T50	Total temperature at LPT outlet
5	P2	Pressure at fan inlet
6	P15	Total pressure in bypass-duct
7	P30	Total pressure at HPC outlet
8	Nf	Physical fan speed
9	Nc	Physical core speed
10	epr	Engine pressure ratio (P50/P2)
11	Ps30	Static pressure at HPC outlet
12	phi	Ratio of fuel flow to Ps30
13	NRf	Corrected fan speed
14	NRc	Corrected core speed
15	BPR	Bypass ratio
16	farB	Burner fuel-air ratio
17	htBleed	Bleed enthalpy
18	Nf_dmd	Demanded fan speed
19	PCNfR_dmd	Demanded corrected fan speed
20	w31	HPT coolant bleed
21	w32	LPT coolant bleed

$$RMSE = \sqrt{\left(\frac{1}{N}\right) \sum_{i=1}^{N} (A_i - F_i)^2} \quad (8)$$

3.4. Implementation

This research provides a new hybrid model that combines CNN and GRU. In this model, pre-processed data are fed into CNN-GRU for prediction. Figure 3 shows a schematic of the suggested framework to help you understand it better. For the data-driven prediction problem, it is important to think about how to put useful time information into the input of the prediction model. If the signal gathered during a sampling period is used as an input sample for the forecast model, the time information in the time series signal is necessarily omitted, limiting the model's prediction performance. To address this issue, this paper employs the time window embedding strategy to process the normalized sensor signal, in which the sensor signal obtained at continuous sampling time steps is concatenated into a high dimensional vector using a fixed-size time window and then inputted into the CNN-GRU model. The shape of images is discussed in this study (30,30). The geometry of the training set, according to 8 selected sensors, is (15631,30,30,8), and the test is (8162,30,30,8).

Fig. 4 Indicates the time sequence of the 8 selected sensor measurements.

Configuration of CNN-GRU- CNN-GRU configuration- Input image data with initial dimension pass through the CNN path. A 1-D convolutional layer is present in this path. After passing through the Maxpooling layer, the output was prepared to be fed into the GRU for learning the model using a Flatten layer. The Mish function has activated all of the neurons in the CNN path.

To extract the properties of the normalized sequence data, a 1D convolutional neural network was developed. Furthermore, gated recurrent unit (GRU) layers collected temporal information from the extracted features. After that, the output features were passed through a fully linked layer for final prediction.

Table 3 shows the network parameters that were chosen. These parameters are determined using several training and testing sets and the cross-validation technique.

RUL prediction and Comparison- The CMAPSS dataset is used in this case to further validate the availability and advantages of the CNN-GRU method in dealing with RUL prediction difficulties.

According to the literature, the early-stage testing point is identified by a constant RUL value of 125 [19]. In this study, the convolution layer and pooling layer serve as feature extraction layers. Second, utilizing the Mish activation function [20] instead of ReLU improves self-adaptation while enhancing prediction accuracy. Fig5. shows the expected and actual RUL of engines.

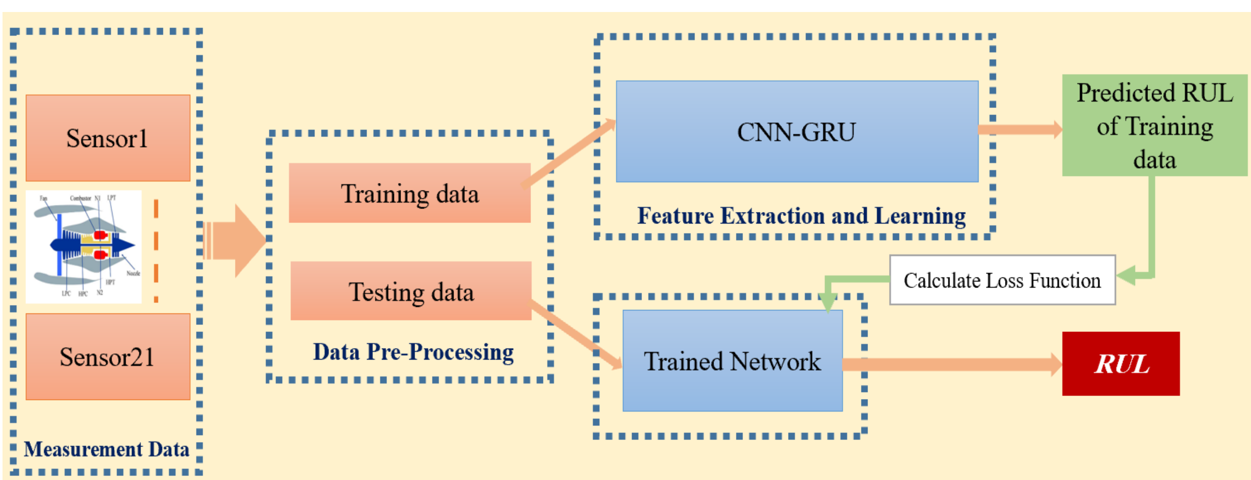

Fig. 3. The proposed procedure.

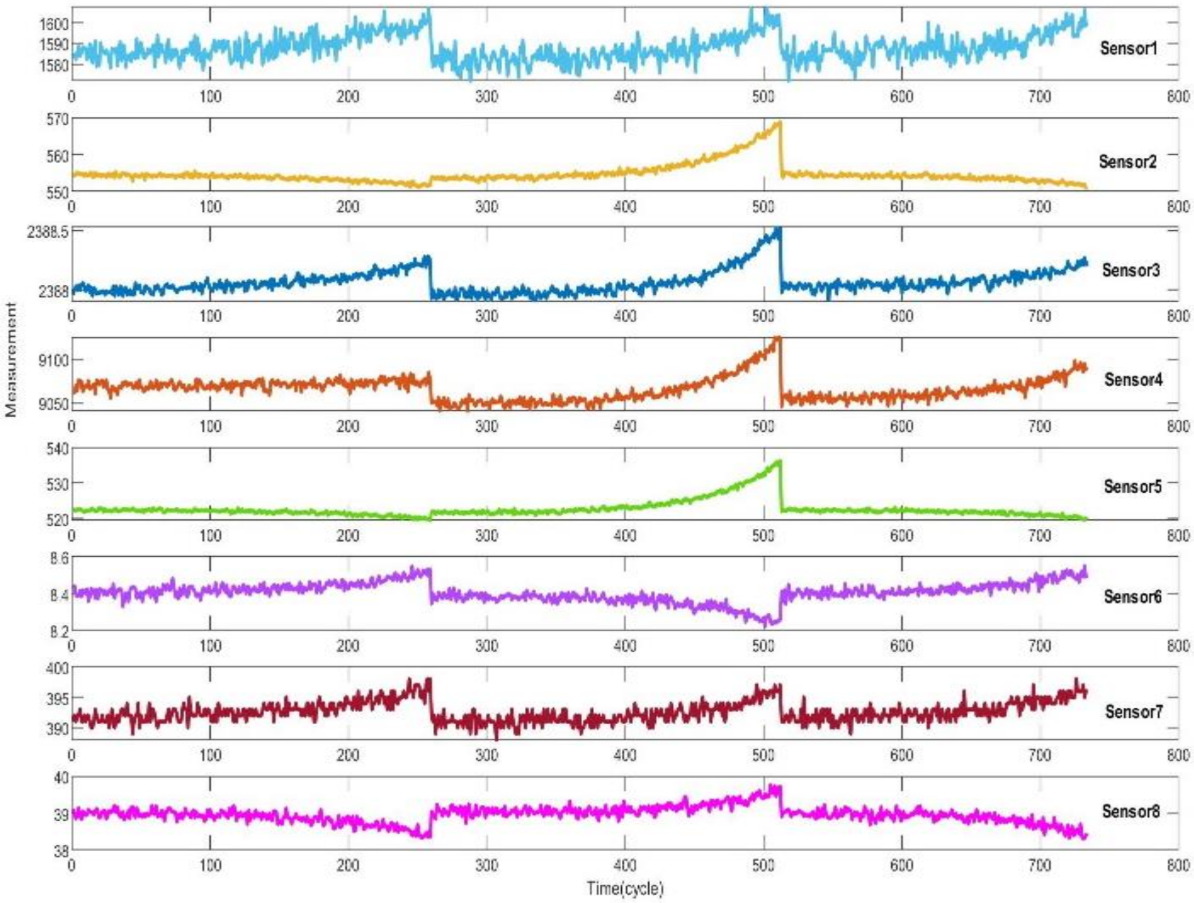

Fig. 4. Sensors' measurements.

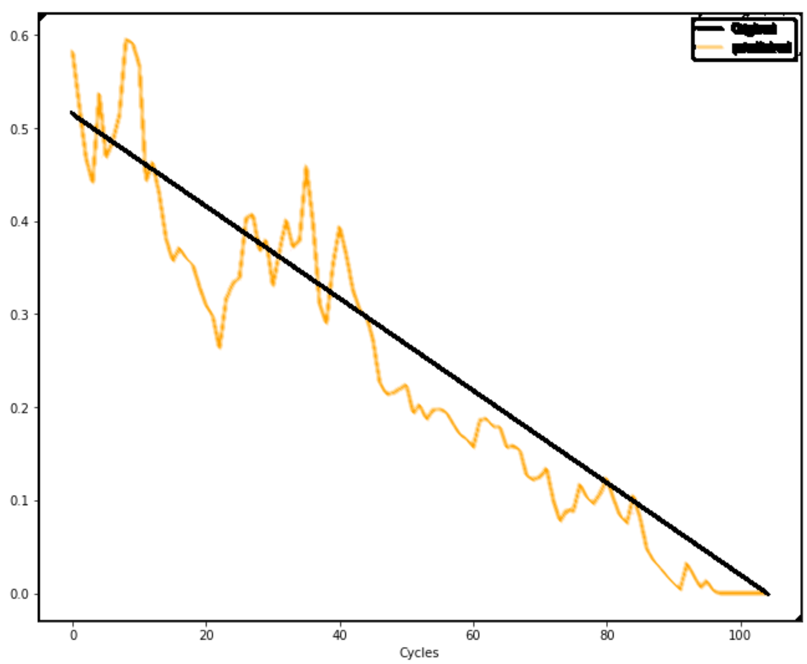

Fig. 5. Predicted and actual RUL.

Table 3. ***Parameters of the Network.***

Parameters	Value
Batch Size	200
Learning Rate	0.001
Epoch	20
Time Window	30

Table 4. ***Different models' performance.***

Model	RMSE
LSTM-CNN [21]	13.01
Concurrent Semi-supervised [22]	12.19
Proposed CNN-GRU	**8.76**

Table 4 summarizes the model performance and compares it to other models. As it is shown, the proposed model has more accuracy compared to the literature.

4. CONCLUTION

In this study, a unique hybrid prognostics approach based on CNN-GRU is used for turbofan engine RUL prediction. Because of the sequential structure of our data, as well as in order to extract the features, the CNN model was employed to extract the features and increase the model's accuracy. In the second stage, the extracted features are entered into the GRU network to train the prediction model.

The proposed model has a single learning and prediction path that includes CNN and GRU. To reduce data to one dimension, a flattened layer is used. On the other hand, the CNN network is used to process the initial dimension of input data. The CNN output is utilized to go through the GRU and develop a model that can predict the RUL.

It should be emphasized that, compared to other prognostics methodologies, the suggested method provides more accurate and robust prediction results than traditional approaches. Furthermore, it was recognized that this will help to reduce maintenance costs and improve maintenance programming.

Because the dataset utilized in this work is based on simulation, we will try our model for the dataset introduced in [23] and based on real flight conditions in future research. Furthermore, digital twins (DT) are becoming increasingly popular among researchers as a powerful tool for prediction. As a result, future work will involve further discussion of additional techniques, including DT, and their benefits in improving the accuracy of our models while decreasing training time.

REFERENCES

[1] L. Peel, "Data driven prognostics using a Kalman filter ensemble of neural network models," in *2008 International Conference on Prognostics and Health Management*, Denver, CO, USA, Oct. 2008, pp. 1–6. doi: 10.1109/ PHM. 2008.4711423.

[2] D. Verstraete, E. Droguett, and M. Modarres, "A Deep Adversarial Approach Based on Multi-Sensor Fusion for Semi-Supervised Remaining Useful Life Prognostics," *Sensors*, vol. 20, no. 1, p. 176, Dec. 2019, doi: 10.3390/s20010176.

[3] O. Bektas, J. A. Jones, S. Sankararaman, I. Roychoudhury, and K. Goebel, “A neural network filtering approach for similarity-based remaining useful life estimation,” *Int J Adv Manuf Technol*, vol. 101, no. 1–4, pp. 87–103, Mar. 2019, doi: 10.1007/s00170-018-2874-0.

[4] L. Guo, N. Li, F. Jia, Y. Lei, and J. Lin, “A recurrent neural network based health indicator for remaining useful life prediction of bearings,” *Neurocomputing*, vol. 240, pp. 98–109, May 2017, doi: 10.1016/j.neucom. 2017.02.045.

[5] [H.-C. Trinh and Y.-K. Kwon, “An Empirical Investigation on a Multiple Filters-Based Approach for Remaining Useful Life Prediction,” *Machines*, vol. 6, no. 3, p. 35, Aug. 2018, doi: 10.3390/machines6030035.

[6] P. R. de O. da Costa, A. Akçay, Y. Zhang, and U. Kaymak, “Remaining useful lifetime prediction via deep domain adaptation,” *Reliability Engineering & System Safety*, vol. 195, p. 106682, Mar. 2020, doi: 10.1016/j.ress.2019.106682.

[7] G. Sateesh Babu, P. Zhao, and X.-L. Li, “Deep Convolutional Neural Network Based Regression Approach for Estimation of Remaining Useful Life,” in *Database Systems for Advanced Applications*, vol. 9642, S. B. Navathe, W. Wu, S. Shekhar, X. Du, X. S. Wang, and H. Xiong, Eds. Cham: Springer International Publishing, 2016, pp. 214–228. doi: 10.1007/978-3-319-32025-0_14.

[8] S. Khan and T. Yairi, “A review on the application of deep learning in system health management,” *Mechanical Systems and Signal Processing*, vol. 107, pp. 241–265, Jul. 2018, doi: 10.1016/j.ymssp.2017.11.024.

[9] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, “Convolutional, Long Short-Term Memory, fully connected Deep Neural Networks,” in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, South Brisbane, Queensland, Australia, Apr. 2015, pp. 4580–4584. doi: 10.1109/ICASSP. 2015.7178838.

[10] H. Yang, F. Zhao, G. Jiang, Z. Sun, and X. Mei, “A Novel Deep Learning Approach for Machinery Prognostics Based on Time Windows,” *Applied Sciences*, vol. 9, no. 22, p. 4813, Nov. 2019, doi: 10.3390/app9224813.

[11] L. Wen, Y. Dong, L. Gao, 1 The State Key Laboratory of Digital Manufacturing Equipment & Technology, School of Mechanical Science & Engineering, Huazhong University of Science & Technology, Wuhan, 430074, China, and 2 School of Electronic Information & Communications, Huazhong University of Science & Technology, Wuhan, 430074, China, “A new ensemble residual convolutional neural network for remaining useful life estimation,” *Mathematical Biosciences and Engineering*, vol. 16, no. 2, pp. 862–880, 2019, doi: 10.3934/mbe.2019040.

[12] Z. Esfahani, K. Salahshoor, B. Farsi, and U. Eicker, “A New Hybrid Model

for RUL Prediction through Machine Learning," *J Fail. Anal. and Preven.*, Jul. 2021, doi: 10.1007/s11668-021-01205-8.

[13] R. Dey and F. M. Salem, "Gate-variants of Gated Recurrent Unit (GRU) neural networks," in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Boston, MA, Aug. 2017, pp. 1597–1600. doi: 10.1109/MWSCAS.2017.8053243.

[14] J. Gu *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, May 2018, doi: 10.1016/j. patcog.2017.10.013.

[15] T. Afridi, A. Alam, N. Khan, and J. Khan, *A Multimodal Memes Classification: A Survey and Open Research Issues*. 2020.

[16] K. Cho *et al.*, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation." arXiv, Sep. 02, 2014. Accessed: Sep. 13, 2022. [Online]. Available: http://arxiv.org/abs/1406.1078

[17] E. R. Abhinav Saxena, "Review and Analysis of Algorithmic Approaches Developed for Prognostics on cmapss dataset," presented at the annual conference of the prognostics and health management society 2014, 2014.

[18] Z. Esfahani, K. Salahshoor, and A. H. Mazinan, "Remaining useful life prognostics based on stochastic degradation modeling: turbofan engine as case study," *J Braz. Soc. Mech. Sci. Eng.*, vol. 44, no. 7, p. 262, Jul. 2022, doi: 10.1007/s40430-022-03546-3.

[19] Y. Wu, M. Yuan, S. Dong, L. Lin, and Y. Liu, "Remaining useful life estimation of engineered systems using vanilla LSTM neural networks," *Neurocomputing*, vol. 275, pp. 167–179, Jan. 2018, doi: 10.1016/j. neucom.2017.05.063.

[20] D. Misra, "Mish: A Self Regularized Non-Monotonic Activation Function," 2019, doi: 10.48550/ARXIV. 1908.08681.

[21] A. Al-Dulaimi, S. Zabihi, A. Asif, and A. Mohammadi, "A multimodal and hybrid deep neural network model for Remaining Useful Life estimation," *Computers in Industry*, vol. 108, pp. 186–196, Jun. 2019, doi: 10.1016 /j.compind.2019.02.004.

[22] T. Wang, D. Guo, and X.-M. Sun, "Remaining useful life predictions for turbofan engine degradation based on concurrent semi-supervised model," *Neural Comput & Applic*, vol. 34, no. 7, pp. 5151–5160, Apr. 2022, doi: 10.1007/s00521-021-06089-1.

[23] M. Arias Chao, C. Kulkarni, K. Goebel, and O. Fink, "Aircraft Engine Run-to-Failure Dataset under Real Flight Conditions for Prognostics and Diagnostics," *Data*, vol. 6, no. 1, p. 5, Jan. 2021, doi: 10.3390/data6010005.