ICMIEE20-231

Remaining Useful Life Prediction of a Motor Gear Test Rig Using Autoencoder and Gated Recurrent Unit

Sultan Ahmed[1], Md. Emdadul Hoque[2], Dip Kumar Saha[1]
[1]Department of Mechatronics Engineering, Rajshahi University of Engineering & Technology, Rajshahi-6204, BANGLADESH
[2]Department of Mechanical Engineering, Rajshahi University of Engineering & Technology, Rajshahi-6204, BANGLADESH

ABSTRACT
In this era of the smart world, it is of utmost importance to predict an upcoming scenario, particularly the failure of a system part in predictive maintenance. A significant term in condition monitoring is the Remaining Useful Life (RUL) of an element, which provides an indication of the time after which the element does not serve. It will result in a more effective way to minimize maintenance costs by forecasting the RUL of a system part with satisfactory accuracy. For forecasting the RUL of a motor gear test rig, this paper proposes a data-driven model. A dataset from the sensors detecting the vibration of the system's bearing in different health stages is fed to a Deep Neural Network (DNN) model, a combination of Autoencoder and Gated Recurrent Unit (GRU). For predicting RUL, the proposed model is faster because GRU consists of only two gates and requires less memory, whereas standard algorithms such as Long Short Term Memory (LSTM) consist of three gates and take more memory to operate on a more significant chain. The Autoencoder combination has made it unique and more effective as it compresses the input data to produce a smaller representation and can be rebuilt to the desired standard level. The model can also be handy for maintenance engineering as it demonstrates promising efficiency relative to conventional time series algorithms.

Keywords: Remaining Useful Life, Predictive maintenance, Long Short Term Memory, Autoencoder, Gated Recurrent Unit.

1. Introduction
Since the invention of technology, machine maintenance has been an inseparable part of it. From the earlier stage, machine maintenance has gone through various stages like [1] periodic or preventive maintenance, corrective or reactive maintenance, predictive or condition-based maintenance. [2] Periodic maintenance is usually done after a definite time interval. Corrective maintenance fixes a machine when it fails to perform. With the development of technology predictive maintenance has now become a topic of huge interest in maintenance engineering. Predictive maintenance uses AI methods for estimating the health condition of a machine which determines that the maintenance is required or not. It is comparatively more cost-effective. Various types of algorithms are used for predicting the Remaining Useful Life (RUL) of a machine component in predictive maintenance. Recurrent Neural Network (RNN) like Long Short Term Memory (LSTM), Gated Recurrent Unit (GRU), combinations of various algorithms are used to predict RUL. This paper proposes a deep neural network of a combination of Autoencoder and GRU to predict the RUL of a motor gear test rig. It shows better accuracy than the existing algorithms.
Predicting RUL of bearing through the AI method is not satisfactory that much due to less degradation and inconsistent data. Wentao et al. [3] proposed an algorithm based on deep feature representation and transfer learning to predict RUL solving this problem. Giduthuri et al. first attempted to use [4] Convolution Neural Network (CNN) for predicting RUL as many of the existing linear models can’t establish a relationship between data and RUL. Linxia et al. [5] combined a data-driven method with a model-based method to estimate the RUL with better accuracy. To prophesy the RUL of a lead-acid battery Chao et al. [6] incorporated the battery's electromechanical model by a particle filter framework. Yanting et al. [7] used Recurrent Neural Network like Long Short Term Memory (LSTM) for predicting the RUL of a supercapacitor. Constrained Kalman filter was used by Junbo et al. [8] for condition monitoring by predicting RUL for the components with a huge amount of noise in the industrial environment. Jun et al. [9] proposed a Deep Long Short Term Memory (DLSTM) network to which time-series signals from multiple sensors were fed to predict the RUL with better accuracy. Theodors et al. [10] proposed an effective data-driven methodology to predict RUL of rolling bearings by incorporating multiple features of the various domains to feed an E-Support Vector Regression (E-SVR) model. Multiscale Convolution Neural Network (CNN) was introduced by Jun et al. [11] for estimating the RUL of a battery. Qibin et al. [12] introduced a signal conversion method to convert a one-dimensional signal into a 2D image applying into a CNN network for predicting RUL. Mingming et al. [13] proposed a combination of support vector machine (SVM) and hybrid degradation tracking model to predict the RUL of a bearing.

2. Recurrent Neural Network
Recurrent Neural Network (RNN) works in a way to predict the next information of a sequence from the previous information. It has a memory part in its structure

* Corresponding author. Tel.: +88-01796405140
E-mail addresses: 27oviahmed27@gmail.com

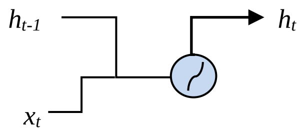

Fig.1 RNN structure [15].

named as hidden state, which stores the information of the previous part of the sequence. The predicted output of a state and all the memory is incorporated as the input to predict the next state of the sequence in the following way [14]

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

$$y_t = \text{softmax}(W_{hy}h_t) \tag{2}$$

Basic RNN [14] faces difficulties due to the gradient problem and exploding problem. Two special types of RNN namely Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) solve these problems. They use two activation functions [15] sigmoid and tanh. The sigmoid function always squishes the values from 0 to 1 [16]

$$g(x) = \frac{1}{1+e^{-x}} \tag{3}$$

The tanh activation function squishes the values from -1 to 1 [16]

$$a = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{4}$$

2.1 Long Short Term Memory

LSTM consists of three gates and a cell state [17] and it is more suitable for solving a long sequence problem.

A. Forget Gate

Forget gate decides which information should be kept and which information should be erased. Forget gate includes sigmoid activation function. The operation [17] performed by the gate is

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{5}$$

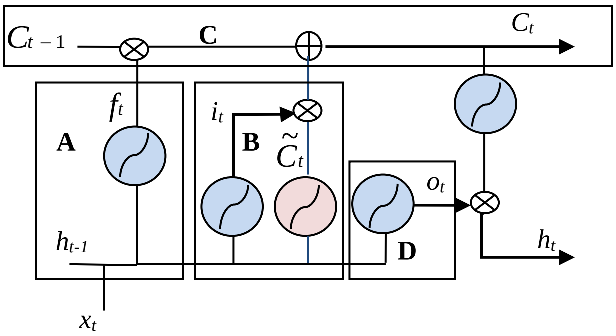

Fig.2 LSTM structure[17].

B. Input Gate

The input gate works to update the cell state or memory. It consists of a sigmoid and a tanh activation function. The previous hidden state (h_{t-1}) and the present input state (x_t) are applied to the two activation functions. The sigmoid output of 0 to 1 decides which information should be kept from the tanh output of -1 to 1 to update the memory. It updates the cell state in the following way [17],

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{6}$$

$$\tilde{C}_t = \tanh(W_c.[h_{t-1}, x_t] + b_c) \tag{7}$$

C. Cell State

Cell state is the pathway of the memory of an LSTM. First of all, some memory is forgotten by the multiplication of the previous cell state (c_t) with the output (f_t) of the forget gate. Then it is updated [17] by adding with the output of the input gate.

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{8}$$

D. Output Gate

The output gate consists of a sigmoid function. The previous hidden state (h_{t-1}) and current input (x_t) are applied to the sigmoid function and then its output is multiplied to get the next hidden state [17].

$$o_t = \sigma(W_0[h_{t-1}, x_t]) + b_0 \tag{9}$$

$$h_t = o_t * \tanh(c_t) \tag{10}$$

2.2 Gated Recurrent Unit

Gated Recurrent Unit [15] performs similarly to LSTM as it is also an RNN. But there are differences in their structure in a way that LSTM consists of three gates whereas GRU consists of two gates. The architecture of GRU is simpler than that of LSTM. As a result, it requires less memory and it runs faster than LSTM in the cases where the dataset is comparatively smaller.

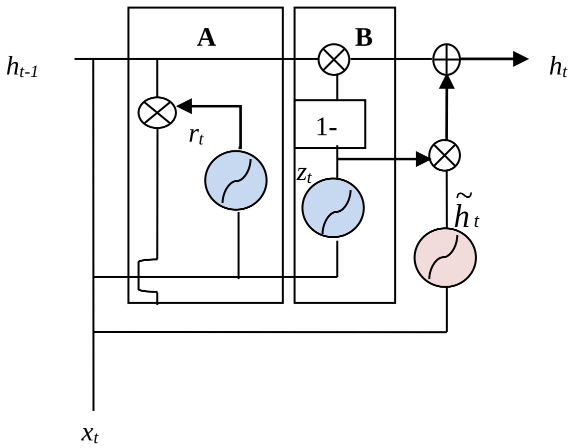

Fig.3 GRU architecture [15].

A. Reset Gate

The reset gate of a GRU consists of a sigmoid activation function. It decides the amount of the memory to forget in the following way [15],

$$z_t = \sigma(W_z[h_{t-1}, x_t]) \tag{11}$$

B. Update Gate

The update gate [15] of GRU is the sum of the forget gate and the input gate of an LSTM. It performs similar applications as the forget gate and input gate jointly do [15].

$$r_t = \sigma(W_r[h_{t-1}, x_t]) \tag{12}$$

$$\tilde{h}_t = \tanh(W[r_t * h_{t-1}, x_t]) \tag{13}$$

$$h_t = (1 - z_t) * (h_{t-1} + z_t * \tilde{h}_t) \tag{14}$$

The existing part of GRU uses tanh activation function. The hidden state (h_t) is the output of the current state and it is also used as the input of the next state.

3. Autoencoder

An autoencoder is an unsupervised technique of Artificial Neural Network (ANN) as it doesn't require any labeled data. To become more accurate, it generates the label of the data itself. The basic purpose of an autoencoder is to produce output the same as the input. But in practical cases due to some losses, it can't generate output the same as the input. It is normally used in the cases of anomaly detection and some generative modeling.

The structure [18] shows that an autoencoder consists of three parts namely encoder, code and decoder. An encoder is an ANN which compresses the input data to reduce its dimension by reducing the number of nodes in every layer. The compressed input is named as code. Then the code is reconstructed by the decoder and the dimensionality of the input is regained. In the decoder, the number of nodes per layer increases according to the reduction of nodes in each layer of the encoder. The architecture of the encoder and the decoder is a mirror to each other.

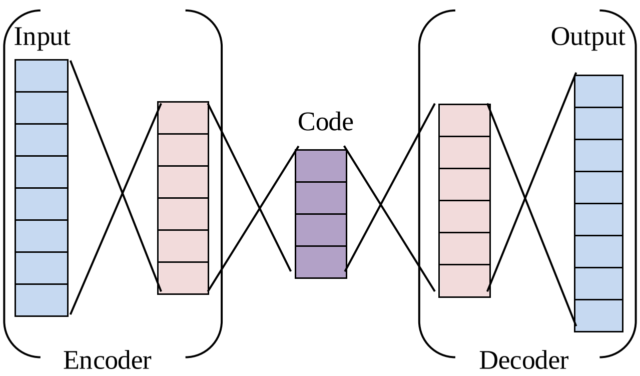

Fig.4 Structure of an autoencoder.

4. Dataset

The PHM08 challenge dataset [19] is used for the proposed algorithm of this paper. It consists of three parts, namely 'Training', 'Test' and 'Final Test'. It represents time series data describing the degradation of a turbo engine which is primarily collected from Commercial Modular Aero-Propulsion System Simulation. The dataset is made from various health conditions of the system.

5. Methodology

The paper proposes a new algorithm to predict RUL by combining autoencoder with GRU. To make the dataset suitable for the model, firstly features are extracted and then it is scaled by using minmaxscaler. As GRU works well with the shorter dataset, the dataset is then applied to an autodencoder model to reduce its dimension. The train and test data with 26 columns are fed to the encoder part of the autoencoder model. The encoder compresses the data and reduces it to 12 columns. Then the compact two-dimensional data is converted into 3D data as the requirement of GRU. A better accuracy has been gained compared to the existing models. Then the probability of failure of the system in 30 days is determined.

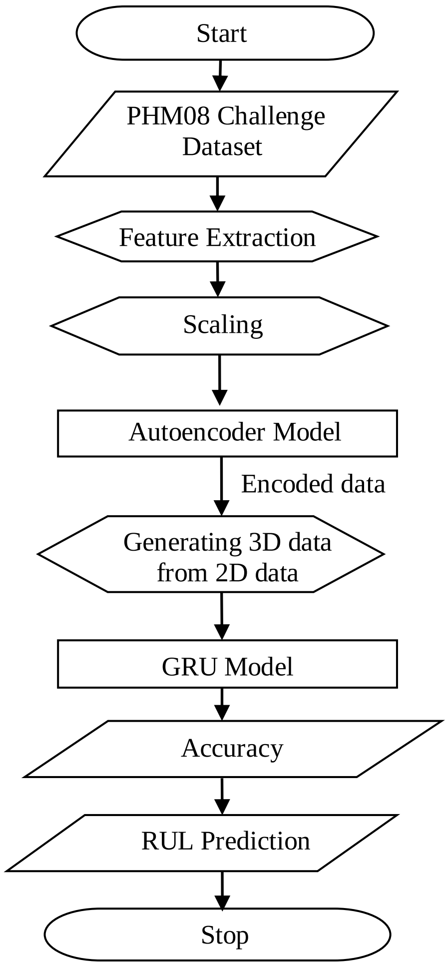

Fig.5 Flowchart of the methodology.

6. Result

A comparative study has been done based on the accuracy of different RNN algorithms and various combinations of algorithms.

Table 1 Accuracy comparison

Serial	Name of the Algorithm	Accuracy (%)	
		Train Data	Test Data
01	LSTM	97.73	98.87
02	GRU	97.86	99.16
03	Autoencoder with LSTM	99.50	99.75
04	Autoencoder with GRU	99.51	99.80

Table 1 summarizes the output of multiple Recurrent Neural Networks. LSTM displays 97.73 percent train accuracy and 98.87 percent test accuracy. But its accuracy improves when Autoencoder is added to an LSTM, as seen in row 3 (train accuracy 99.50 percent and test accuracy 99.75 percent). Combining Autoencoder thus illustrates increased accuracy. The proposed model has improved the training accuracy from 97.86 percent to 99.51 percent and test accuracy from 99.16 percent to 99.80 percent by integrating Autoencoder with GRU. In both train and test cases, the efficiency of the Autoencoder combination with GRU is, therefore, higher than the other RNN algorithms.

7. Conclusion

A new algorithm was built in this article, integrating Autoencoder with GRU to predict RUL with greater precision than the current models using NASA's turbofan engine dataset. It produces 99.51 percent for train data and 99.80 percent for test data, the best accuracy. The loss of the model is pretty low (2%) compared to other models.

8. References

[1] Zhu, K., X., Sensor-based Condition Monitoring and Predictive Maintenance-An Integrated Intelligent Management Support System," *Intell. Syst. Accounting, Financ. Manag.*, vol. 5, no. 4, pp. 241–258, 1996.

[2] Saha, D., K., Ahmed, S., Shaurov, M., S., Different Machine Maintenance Techniques of Rotary Machine and Their Future Scopes: A Review, In: proceedings of 4th Int. Conf. Electr. Inf. Commun. Technol. EICT 2019, 20–22 December, Khulna, Bangladesh, 2019.

[3] Mao, W., He, J., Zuo, M., J., Predicting Remaining Useful Life of Rolling Bearings Based on Deep Feature Representation and Transfer Learning, *IEEE Trans. Instrum. Meas.*, vol. 69, no. 4, pp. 1594–1608, 2020.

[4] Navathe, S., B., Wu, W., Shekhar, S., Du, X., Wang, X., S., Xiong, H., "Database Systems for Advanced Applications: 21st International Conference, DASFAA 2016 Dallas, TX, USA, April 16–19, 2016 Proceedings, Part I," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9642, pp. 214–228, 2016.

[5] Liao, L., Köttig, F., A hybrid framework combining data-driven and model-based methods for system remaining useful life prediction, *Appl. Soft Comput. J.*, vol. 44, pp. 191–199, 2016.

[6] Lyu, C., Lai, Q., Ge, T., Yu, H., Wang, L., Ma, N., A lead-acid battery's remaining useful life prediction by using electrochemical model in the Particle Filtering framework, *Energy*, vol. 120, pp. 975–984, 2017.

[7] Zhou, Y., Huang, L., Pang, J., Wang, K., Remaining useful life prediction for supercapacitor based on long short-term memory neural network, *J. Power Sources*, vol. 440, no. August, p. 227149, 2019.

[8] Son, J., Zhou, S., Sankavaram, C., Du, X., and Zhang, Y., Remaining useful life prediction based on noisy condition monitoring signals using constrained Kalman filter, *Reliab. Eng. Syst. Saf.*, vol. 152, pp. 38–50, 2016.

[9] Wu, J., Hu, K., Cheng, Y., Zhu, H., Shao, X., Wang, Y., Data-driven remaining useful life prediction via multiple sensor signals and deep long short-term memory neural network, *ISA Trans.*, vol. 97, no. xxxx, pp. 241–250, 2020.

[10] Loutas, T., H., Roulias, D., Georgoulas, G., Remaining useful life estimation in rolling bearings utilizing data-driven probabilistic E-support vectors regression, *IEEE Trans. Reliab.*, vol. 62, no. 4, pp. 821–832, 2013.

[11] Zhu, J., Chen, N., Peng, W., Estimation of Bearing Remaining Useful Life Based on Multiscale Convolutional Neural Network, *IEEE Trans. Ind. Electron.*, vol. 66, no. 4, pp. 3208–3216, 2019.

[12] Wang, Q., Zhao, B., Ma, H., Chang, J., Mao, G. , A method for rapidly evaluating reliability and predicting remaining useful life using two-dimensional convolutional neural network with signal conversion, *J. Mech. Sci. Technol.*, vol. 33, no. 6, pp. 2561–2571, 2019.

[13] Yan, M., Wang, X., Wang, B., Chang, M., Muhammad, I., Bearing remaining useful life prediction using support vector machine and hybrid degradation tracking model, *ISA Trans.*, vol. 98, no. xxxx, pp. 471–482, 2020.

[14] Vinyals, O., Ravuri, S., V., Povey, D., REVISITING RECURRENT NEURAL NETWORKS FOR ROBUST ASR International Computer Science Institute , Berkeley , CA , USA EECS Department , University of California - Berkeley , Berkeley , CA , USA," *Network*, pp. 4085–4088, 2012.

[15] Kumar, S., Hussain, L., Banarjee, S., Reza, M., Energy Load Forecasting using Deep Learning Approach-LSTM and GRU in Spark Cluster, *Proc. 5th Int. Conf. Emerg. Appl. Inf. Technol.*

EAIT 2018, pp. 1–4, 2018.

[16] Farzad, A., Mashayekhi, H., Hassanpour, H., A comparative performance analysis of different activation functions in LSTM networks for classification," *Neural Comput. Appl.*, vol. 31, no. 7, pp. 2507–2521, 2019.

[17] Li, J., Li, X., He, D., A Directed Acyclic Graph Network Combined With CNN and LSTM for Remaining Useful Life Prediction, *IEEE Access*, vol. 7, no. c, pp. 75464–75475, 2019.

[18] Ren, L., Sun, Y., Cui, J., Zhang, L., "Bearing remaining useful life prediction based on deep autoencoder and deep neural networks," *J. Manuf. Syst.*, vol. 48, no. November 2017, pp. 71–77, 2018.

[19] Bektas, O., Jones, J. A., Sankararaman, S., Roychoudhury, I., Goebel, K., Reconstructing secondary test database from PHM08 challenge data set," *Data Br.*, vol. 21, pp. 2464–2469, 2018.