A Two-Stage Machine-Learning-Based Prognostic Approach for Bearing Remaining Useful Prediction Problem

Dongdong Zhao and Liu Feng

***Abstract*—In the studies of machinery remaining useful life (RUL) prediction, the construction of health indicator (HI) can appropriately describe operating condition of machinery, which is always the bridge between raw data and RUL. In this paper, an improved HI construction method is utilized to solve glitches introducing when outlier region correction, which are adaptive generalized framework of HI construction and can apply to vibration-signal-related scenarios. On the basis of it, gated recurrent units (GRUs) are proposed to investigate the degradation information over time aim to output a estimated RUL as close to actual RUL as possible. The proposed method in this paper is compared against the state-of-the-art. For further verification, we have choose two dataset come from varying platform when experimental verification. At the same time, long short-term memory (LSTM) also is the one of baseline, which makes recurrent-neural-networks-based (RNN-based) architectures identified as an effective method for RUL prediction. Experimental verification has been carried out on FEMTO and XJTU-SY dataset, experimental results testify proposed method outshines other baselines.**

***Index Terms*—PHM, health indicator, remaining useful life prediction, gated recurrent units**

I. Introduction

PROGNOSTICS health management (PHM) is one of the core research hotpots in the field of mechanical engineering. The degradation and break-down of components seriously affects operating efficiency and profit, and even causes greater loss in life and property. Therefore, there are lots of researches in PHM, for example, generative adversarial networks (GAN) are used in fault diagnosis on few fault samples [1], cross-domain adaption for remaining useful life (RUL) prediction [2]. Also, many RUL studies are proposed in recent years, which classified into two categories: one-stage/end-to-end and two-stage. One-stage methods mean that RUL are directly estimated from a model trained with the input of raw vibration data. Two-stage methods usually firstly construct health indicator (HI) and secondly estimate RUL which can complete main regression estimation tasks with high quality. From the literature in recent years, the latter is more, that is, studies on two-stage methods for RUL estimation are many.

In the rise of artificial intelligence (AI), huge breakthroughs have been made in many domains, especially in autonomous driving, natural language processing (NLP) and computer vision (CV). In the field of bearing PHM research, traditional prognosis algorithms usually does not have considerable generalization ability and anti-interference ability, such as traditional and simple time domain features and frequency domain features. However, for many deep learning (DL) techniques, such as convolutional neural networks (CNN) and long short-term memory (LSTM) have been validated that outperforms other traditional algorithms [3] [4] [5] [6]. For CNN has a strong advantage in nonlinear feature extraction, many scholars use CNN to extract vibration features of bearings. [7] used the CNN with multi-scale filters to construct HI. [8] proposed double-convolutional neural network to predict RUL and got a great performance. [9] applied deep convolutional neural networks (DCNN) extract multi-scale features from time-frequency domain. [4] [10] also utilized CNN and got effectively visible results.

From the perspective of mechanical degradation issue on bearing units, it is a problem of nonlinear degradation over time. For CNN, It is usually difficult to obtain information about time series. Of course, it works when more neural network layers are used to ensure the receptive field large enough, but it will cause some troubles in calculation. Recurrent neural networks (RNN) combining with CNN for RUL prediction are proposed to alleviate the problem of incapable of parallel computing [11]. Because RNN has two defects that cannot remember the longer knowledge and vanishing gradient or exploding gradient, researchers [12] added input and output gates on the basis of RNN to replace the internal logic calculation method to better control input and output streams, which makes LSTM an effective tool related to timing. [13] utilized bidirectional handshaking LSTM to extract degradation information related to time. LSTM is also used to acquire trend of engine for a good score [14]. However, in many practical applications, real-time and accuracy should be first considered, such as NLP and autonomous driving, in order to further reduce training time and improve performance, gated recurrent units (GRUs) are proposed to cover their shortcomings. [15] used GRUs to predict RUL on the basis of nonlinear feature obtained by kernel principle component analysis (KPCA), which is verified and obtained excellent performance on two data sets. Since KPCA requires the calculation of the kernel function,

Manuscript received January 12, 2021; revised September 15, 2021. This work was supported by the National Key RD Program of China under grant No.2019YFB2102500 and the Science Foundation of Beijing-Shanghai High Speed Railway Co.,Ltd under grant No.2019166.(Corresponding author: Dongdong Zhao.)

D. Zhao is a PhD candidate of Beijing Jiaotong University, No.3 Shangyuancun Haidian District, Beijing, China, 100044 (e-mail: ddzhao@bjtu.edu.cn).

F. Liu is a Professor of Computer and Information Technology Department, Beijing Jiaotong University, No.3 Shangyuancun Haidian District, Beijing, China, 100044 (e-mail: fliu@bjtu.edu.cn).

which means a large amount of calculation. In this paper, we obtain one dimensional HI extracted from raw vibration signal via CNNs, then GRUs are used for learning the law of changes over time. In short, a novel two-stage approach for RUL prediction is proposed in this paper where three main contributions are listed as follows:

(1) HI construction. Based on [16], we optimize outlier region correction in a local perspective instead of in a global perspective, because the outlier region correction method based on global will introduce some glitches. This will provide quality data with less noise for subsequent sequence learning.

(2) RNN-based RUL prediction. On the basis of (1), GRUs are used to extract degradation information of sequential HIs.

(3) To make sure the effectiveness of the proposed method, we do a lot of experiments on two data sets.

The organization of the rest of this article is as follows. Section II will use related work as a foreshadowing: health indicator construction, standard RNN, LSTM, and GRU. Section III will give out experimental setting: dataset, data preprocessing, training precedure and hyper-parameter selection. Section IV shows experimental results. Finally, conclusion in section V closes this paper.

II. Related Works

A. Health Indicator Construction

HI construction usually be grouped into two classes: physical HIs and fused HIs [16]. Many methods are still belong to handcrafted feature that faces some problems that HIs construction depends on knowledge of the characteristic of the acquired data. [16] utilized CNN to automatically build HIs where the architecture includes one input layer, two convolutional layers, two pooling layers and three fully connected layers, also named CHI. In the final fully connected layer, the output is a HI corresponding to cycle time step t, by minimizing the square of the euclidian distance between output value and the actual degradation percentage corresponding to the vibration signal input X_t = (x_1,x_2,...,x_n) of cycle time t, n is the number of data points.

For a bearing unit, after HIs constructed in whole cycle time, outlier region correction is utilized to remove the outlier regions that are determined by 3σ rule, which is used in status monitoring [9] [17] [18]. In this rule, any point with HI value greater than μ+3σ or less than μ-3σ is judged to be an outlier, which affects the true trend of HIs like noise. Readers can refer to [16] for detailed information about CHI. In the following chapters, we will give an optimization plan for CHI, and design a reasonable experimental verification.

B. Standard RNN

Compared with fully connected neural networks, RNN has one more hidden unit. RNN structurally connects multiple neural hidden units in series show in Fig.1. Its input is a string of data X_t=(x_{t-L+1},x_{t-L+2},...,x_t)^T, the length L of X in current time t is same as the number of hidden units, and the entire RNN structure shares one group (U, W, b). W is a matrix that bridges the current hidden layer and the previously hidden layer. different with W, U is the connection matrix linking the current input layer and hidden layer. b is the bias on the hidden unit. Each time the preceding hidden unit is calculated, the corresponding state factor is learned, which provides knowledge support for the subsequent hidden units. For each time step t, the update function hidden state and output y_t at time step t as follows,

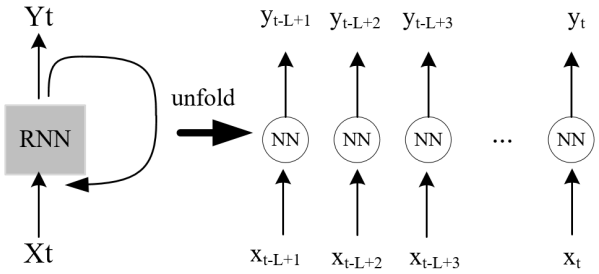


Fig. 1: The architecture of standard RNN.

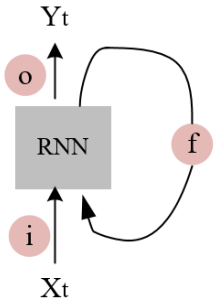


Fig. 2: The whole architecture of LSTM.

$$\begin{aligned} h_t &= f(Wh_{t-1} + Ux_t + b), \\ y_t &= f(Vh_t) \end{aligned} \tag{1}$$

, where V is a matrix that notes the relationship between the current hidden layer and output layer. Because of parameter sharing, when propagating gradient optimization after training loss is calculated, the loss is continuously multiplied by a number less than 1, which will cause the loss to become a number close to 0 before it is passed to the pre-order hidden layer. Therefore, the parameters of the previous hidden layer cannot be updated, which makes the difficult in training. Conversely, if multiplied by a value greater than 1, the gradient explodes. To solve these shortcomings of RNN, LSTM is proposed in [12].

C. LSTM

LSTM has no difference between external input and output with RNN. Instead, the internal update calculation method is replaced by input gate i, forget gate f, output gate o and internal memory unit c, the schematic diagram is shown in Fig.2.

The logic operation process is described in Fig.3. Forgetting gate controls the degree to which the input x and the output h of the previous hidden layer are forgotten.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_r) \tag{2}$$

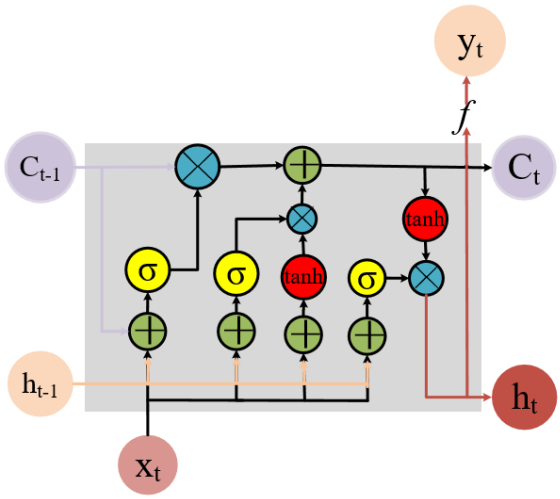


Fig. 3: The internal logical operation of LSTM.

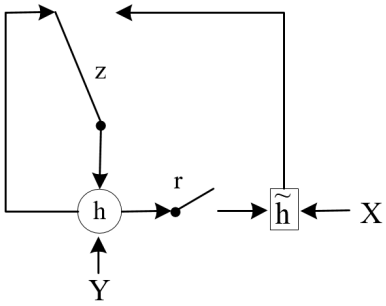


Fig. 4: The architecture of GRU.

Input gate controls the extent to which the input x and the current calculated state are updated to the memory unit.

$$\begin{aligned} i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ \tilde{C}_t &= tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \end{aligned} \tag{3}$$

Output gate controls input x and the current output gate depending on the degree of the current memory unit.

$$\begin{aligned} o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o), \\ h_t &= o_t * tanh(C_t), \\ \hat{y}_t &= f(h_t) \end{aligned} \tag{4}$$

where f is a activation function. $\hat{y}_t$ is the final output of LSTM. These three gates respectively share their own (U, W, b), which can better control the inflow and outflow of information so that the entire network can better get the relationship between sequence information.

D. GRU

Because the LSTM gate control network structure is too complicated and redundant we can see from previous content about LSTM, GRU reduces the number of gates from three to two: reset gate and update gate shown in Fig.4 . GRU introduces an update gate to replace the forget gate and input gate and introduces a reset gate to replace the memory unit and the hidden unit, which makes the entire logic operation more concise and performance enhanced. The function reset gate r_t:

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{5}$$

TABLE I: Information of FEMTO Dataset

Condition	Load(N)	Speed(rpm)	Name
1	4000	1800	A1-1, A1-2, A1-3, A1-4, A1-5, A1-6, A1-7
2	4200	1650	A2-1, A2-2, A2-3, A2-4, A2-5, A2-6, A2-7
3	5000	1500	A3-1, A3-2, A3-3

The function of update gate z_t:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{6}$$

The hidden state is calculated from reset gate $\tilde{h_t}$:

$$\tilde{h_t} = tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \tag{7}$$

The final hidden state h_t is updated by update gate:

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h_t} \tag{8}$$

The task of GRU is to map the current health indicator to a RUL value, and the proposed method uses the sequence-to-one method. The whole life cycle data of each bearing unit is divided into fixed-length sequences X_t = (x_{t-L+1},x_{t-L+2},...,x_t), L is the length of the time series, x_t is the health indicator at the current time t. The current actual RUL value is y_t. First, X_t and y_t are fed into the GRU to obtain a predicted RUL. The Mean Absolute Error (MAE) between the predicted RUL and the corresponding actual RUL is used as the loss, and the model parameters are updated through the gradient descent sgd [19] method.

$$MAE = \frac{1}{N}\sum_{n=1}^{N}|\hat{y} - y|, \tag{9}$$

where N is the number of cycle time of a bearing unit. If the MAE value of the training set is no longer updated more than 20 iterations, stop training.

III. Experimental Setting

A. Dataset Description

1) FEMTO dataset: FEMTO dataset, collected from the platform PRONOSTIA, is aimed to investigate bearing unit degradation through conducting degradation experiments and obtaining vibration data from monitors of bearing unit. The FEMTO dataset includes 3 operation conditions, which respectively has 7, 7 and 3 bearing units, and the raw vibration signals are the run-to-failure data. For every cycle time step, there are 2560 data points in horizontal direction with sampling frequency of 25.6 kHz. Simple information of the FEMTO dataset are shown in Table I. Ax-y represents the y-th bearing unit under the x-th condition of FEMTO dataset. For more detailed information of the dataset, readers can refer [20].

2) XJTU-SY dataset: XJTU-SY dataset, provided by the Xi'an Jiaotong University and the Changxing Sunyoung Technology company [21], is also degradation tests of bearing units for the investigation of health condition monitoring and RUL prediction. This dataset was also collected with the same sampling frequency as the FEMTO dataset. The difference is that each sample includes 32768 data points. The simple information are shown in Table II. Bx-y represents the y-th bearing unit under the x-th condition of XJTU-SY dataset. For more detailed information, researchers can refers to [21].

TABLE II: Information of XJTU-SY Dataset

Condition	Load(N)	Speed(rpm)	Name
1	12000	2100	B1-1, B1-2, B1-3, B1-4, B1-5
2	11000	2250	B2-1, B2-2, B2-3, B2-4, B2-5

B. Data Preprocessing and Performance Metrics

In experimental verification process, one bearing unit is randomly selected as the test data, and the remaining bearing units constitute training data. What needs further explanation is that the condition difference is ignored in the random test bearing selection process, which means that this experimental study does not distinguish the bearing operating conditions, but treats all bearings equally, thereby improving the generalization ability of the algorithm and the actual practicality in application. This mechanism of experimental data construction is also used in [16].

In order to make the dimension number of the two datasets same with each other, the data are subjected to coarse-grained physical dimensionality reduction before experimental verification. By this process, the algorithm's dependence on high-frequency vibration data is also reduced as much as possible. Reducing the data dimension can better adapt to actual situation, which means proposed method can backward compatible to equipment with a low configuration vibration signal collector.

In the building process of sequence data construction, if current cycle time step t_c is less than the sequence length L, a sequencial data with fixed length L will be supplemented to by zeros.

During the test process, the GRU model outputs a estimated RUL between 0 and 1, which needs to be converted into an actual value. The conversion formula follows the this formula:

$$\hat{RUL} = \frac{\hat{y}_t * R_t}{y_t}, \tag{10}$$

where y_t is actual label, $\hat{y}_t$ is the predicted RUL, and R_t is the real RUL corresponding to the current moment t. There is a actual RUL R_t corresponding to current cycle time step t of tested bearing unit. If the total cycle time is 50000 s and the current time is 20000 s, then R_t is 30000 s.

Different metrics measures the performance of an algorithm from different prospective. When verification, MAE and root mean square error (RMSE) are participated in calculating experimental accuracy. The calculation formula of MAE and RMSE is as follows:

$$\begin{aligned} MAE &= \frac{1}{N}\sum_{n=1}^{N}|\hat{RUL} - RUL|, \\ RMSE &= \sqrt{\frac{1}{N}\sum_{n=1}^{N}(\hat{RUL} - RUL)^2} \end{aligned} \tag{11}$$

Where RUL is the actual RUL value, and the $\hat{RUL}$ is the estimated RUL value, N is the number of cycle time of bearing unit.

C. Training Procedure and Hyper-parameter Selection

Experimental training procedure is listed as the following steps:

TABLE III: Parameters in HI Construction Stage and RUL Prediction Stage.

Parameters	value	Parameters	value
lr_{HI}	0.001	lr_{RUL}	0.001
$dropout$	0.5	L	20
$optimizer$	sgd	lf	MAE

1) Data preparation: FEMTO/XJTU-SY dataset are grouped into training data and test data.

2) Obtain health indicator of training data and test data: Firstly, initialize CHI model like [16] and training data are passed it for training the model. Secondly, some outlier region and glitches of HI corresponding to training data or test data are rectified by an improved outlier region correction method for final HI of the training data HI_{tr} and of the test data HI_{te}.

3) Training GRU model: Initial the GRU model, use HI_{tr} to train GRU model and save model parameter.

4) Verification and performance calculation: Load trained GRU model, predict RUL of the HI_{te}, and calculate RMSE and MAE between predictive RUL and actual RUL.

In the HI construction stage, [16] has found the best parameters pair through experimental verification. To further confirm the generalization ability of this proposed method, which is carried on the XJTU-SY data set and got good performance. The parameters of CHI construction stage are shown in Table III. In RUL prediction stage, in order to explore the impact of different L, we have carried some tests through single-varible method for searching a favorable length L of GRU.

IV. Experimental Results

A. Health Indicator Construction

CHI includes two stages: HI construction and outlier region correction. In the process of outlier region correction, we find when the time span between the first outlier point and the last outlier point is large, some small glitches will appear. The outlier correction formula in [16] is as follows:

$$h^i_{t_c} = h^c_{t_s} + \frac{h^c_{t_e} - h^c_{t_s}}{t_e - t_s}(t_c - t_s) \tag{12}$$

Where $h^i_{t_c}$ is the CHI at time t_c, t_e and t_s are respectively the time of first outlier and last outlier. The perspective is global. For example as Fig.5 shown, CHI in [16] will introduce some glitches when constructing HIs. In order to solve the issue, we improved this method in a local perspective. The step to remove the outlier regions uses the variable gap representing the cycle time difference d between two adjacent outliers O_i and O_{i+1}, $i \in [1, N-1]$, N is the total number of positive and negative outliers, O_i is the i-th outlier. If d is less than the gap, we think that the span between two adjacent outliers is small, and all points between them will be removed as shown in Fig.5.

$$h^i_{t_c} = h^c_{t_{O_i}} + \frac{h^c_{t_e} - h^c_{t_s}}{t_e - t_s}(t_c - t_{O_i}) \tag{13}$$

Where $h^c_{t_{O_i}}$ is the outlier O_i at time t_{O_i}, $h^i_{t_c}$ is the CHI at time t_c, $t_c \in [t_{O_i}, t_{O_{i+1}}]$.

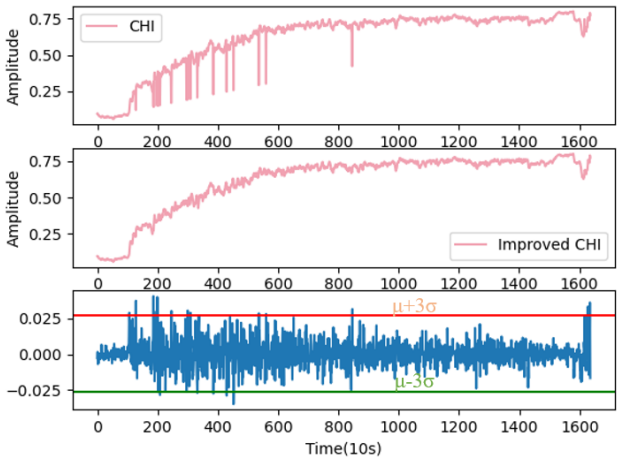


Fig. 5: Comparison of health indicators construction before and after optimization on FEMTO dataset.

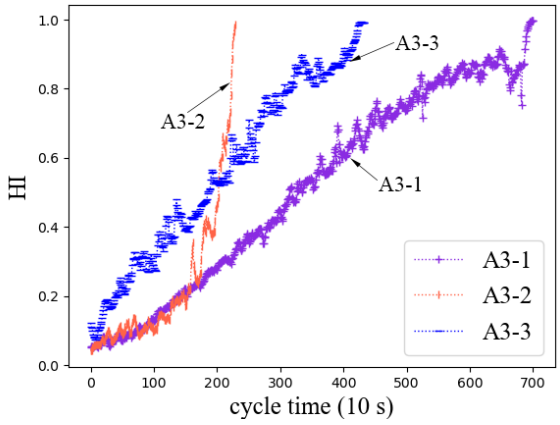


Fig. 6: Health indicator construction on FEMTO dataset.

Otherwise, if d is greater than the gap, we think that the span between two adjacent outliers is large which will introduce some glitches. For this problem, we ignore the adjacent outliers and continue to detect if next pair of adjacent outilers need to be carried on removing untill all the pairs are traversed. In this paper, we also let gap equals to 'Length' in [16].

On the whole, improved CHI method uses local initial outlier instead of global initial outlier, although keeping the slope constant. In Fig.5, the top sub-figure in Fig.5 comes from CHI [16], the middle sub-figure in Fig.5 comes from improved CHI in this paper, the bottom sub-figure in Fig.5 is the difference of HI. Because 3σ principle is aimed at outliers that are less than 3% out of range. So the outlier region correction has far-reaching significance for real-time condition monitoring and RUL estimation, because the improved method reduces the interference with subsequent RUL estimation. Fig.6 and Fig.7 respectively shows HI construction of some bearing units from two datasets which all maintain monotonicity in the overall situation, although there are small fluctuations in the local area, which provides a good reference basis for the subsequent RUL prediction.

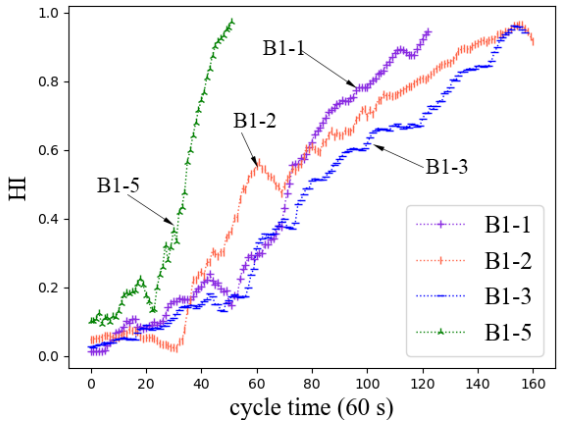


Fig. 7: Health indicator construction on XJTU-SY dataset.

TABLE IV: RUL Prediction Error with GRU in Different L on FEMTO Dataset.

L	8	16	24	32
$RMSE$	40.92	21.77	19.11	16.27
MAE	29.28	15.29	13.49	12.16

B. Remaining Useful Life Prediction

The number L of hidden neural units that represents the depth of GRU directly influences the performance of GRU. In order to explore the impact of L on GRU performance, 10 repeated experiments with different L at 8, 16, 24 and 32 under same other parameters are carried out and the average results are also shown in Table IV and Table V. From Table IV, it is easy to know that the longer sequence length is, the smaller prediction error is. In [15], experimental results turn that the prediction error is negatively corrected with sequence length. The sample number of bearing on XJTU-SY is less than FEMTO's in spite of bearing units on XJTU-SY dataset has a larger time unite, the smallest sample number of bearing on XJTU-SY is 42. From Table V, the proposed method will obtain a lower MAE when parameter L equals 32 though can get a lower RMSE when parameter L equals 16, which may be come from shorter samples on XJTU-SY dataset. But as we all known, we are willing to choose L=16 when a run-to-failure bearing unit has a few samples.

The abscissa and ordinate of Fig.8 and Fig.9 respectively represent cycle time (sample number) and RUL percentage. The unit length of the abscissa is 10 second in Fig.8 and 60 second in Fig.9, and the ordinate represents the RUL percentage.

Fig.8 shows the prediction results of three bearing units under different operating conditions in the FEMTO dataset. A 95% confidence interval is given in each sub-graph. Judging from the bearing unit of each operating condition, the prediction results almost all fall within the

TABLE V: RUL Prediction Error with GRU in Different L on XJTU-SY Dataset.

L	8	16	24	32
$RMSE$	23.65	22.68	24.50	23.44
MAE	19.74	18.33	19.96	17.25

TABLE VI: RUL Prediction Error with Different Methods on FEMTO Dataset.

	GRU		LSTM		XGB		DG	
	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
A1-1	20.23	13.76	23.90	16.69	34.32	19.87	36.45	27.29
A1-2	5.70	4.57	7.73	5.79	9.65	5.99	11.14	8.26
A1-3	35.19	29.63	37.24	30.95	38.54	33.45	38.99	35.66
A1-4	17.65	13.17	17.80	14.48	25.42	16.92	17.82	13.72
A1-5	15.24	12.4	15.24	12.54	18.18	12.43	26.77	20.24
A1-6	10.6	7.60	11.77	9.61	19.67	10.77	19.56	12.07
A1-7	23.35	17.42	24.08	17.62	30.36	19.30	37.23	25.36
A2-1	6.65	5.50	7.87	5.99	33.6	12.62	33.35	15.15
A2-2	6.08	4.72	6.44	5.04	14.49	8.80	14.78	10.89
A2-3	36.01	24.12	38.56	24.82	44.19	25.47	39.03	25.21
A2-4	5.76	4.70	5.66	4.67	11.34	7.58	12.70	8.88
A2-5	43.85	29.01	50.84	33.95	71.15	34.53	73.51	41.56
A2-6	9.29	7.62	10.41	8.73	3.29	8.49	16.37	11.69
A2-7	8.72	6.75	10.53	9.14	12.41	7.42	12.82	7.81
A3-1	5.14	4.27	6.65	5.67	6.88	4.43	7.26	5.54
A3-2	26.14	20.62	27.22	20.75	27.99	22.35	27.50	20.98
A3-3	1.14	1.00	1.24	1.12	2.44	1.49	1.58	1.13
Ave	16.27	12.16	17.83	13.38	23.76	14.81	25.10	17.14

TABLE VII: RUL Prediction Error with Different Methods on XJTU-SY Dataset.

	GRU		LSTM		XGB		DG	
	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
B1-1	12.41	9.10	20.65	17.81	26.01	18.59	21.44	18.81
B1-2	19.71	15.18	32.91	18.44	23.44	26.17	27.19	22.16
B1-3	30.86	22.22	50.61	43.62	34.85	25.67	33.72	27.85
B1-4	11.04	8.01	20.95	17.97	33.21	24.04	23.06	19.44
B1-5	4.69	3.40	7.02	5.80	11.76	8.30	7.48	6.39
B2-1	41.25	30.63	43.73	33.30	89.13	66.23	57.16	46.74
B2-2	28.23	20.74	43.42	27.45	39.67	29.67	32.52	37.00
B2-3	53.79	38.47	51.12	36.57	56.58	38.75	58.67	44.50
B2-4	7.87	6.58	8.79	7.42	7.49	5.12	7.92	6.98
B2-5	24.61	18.18	25.80	19.17	43.78	29.98	34.12	23.24
Ave	23.44	17.25	30.50	22.75	35.59	27.25	30.32	25.31

interval. Fig.9 shows the prediction results on the XJTU-SY dataset. For the B1-2 bearing unit shown in Fig.9(a), a few prediction values are outside the confidence interval, but as time goes by, the prediction values are closer to the true values. From Fig.8 and Fig.9, we can see the proposed solution is capable of predicting RUL and has the generalization ability of RUL estimation of bearing units under different machines.

For further comparing, experimental baseline consists of xgboost (XGB), double gaussian (DG) and LSTM. XGB has repeatedly obtained excellent results in data classification and regression of science competition. DG are used to predict RUL and get a good result in [11]. According to DG and XGB, we choose best parameters through grid search technology. To ensure a fair comparison, LSTM has same parameters with GRU's, such as number L of hidden neural units.

Table VI and Table VII show RMSE and MAE of RUL prediction on the FEMTO dataset and XJTU-SY dataset respectively. In these two tables, the average RMSE and MAE of GRU are lower than other baselines. Compared with XGB and DG, the RMSE and MAE of LSTM are closer to GRU's, which proves that RNN-based is suitable to solve the nonlinear degradation problem.

V. Conclusion

In this research, the improved technology solves the glitches introduced during the correction of abnormal regions. Based on the construction of health indicators, we introduce GRU to extract nonlinear degradation features and predict the estimated RUL value. After a lot of experimental verification, from the perspective of RMSE and MAE, the combination of CHI and GRU works very well in the mechanical RUL prediction of the FEMTO data set and the XJTU-SY data set.

In the future works, we are focusing on cross-condition and cross-machine RUL prediction, which urgently needs more scholars in related fields to study.

References

[1] D. Zhao, F. Liu, and H. Meng, "Bearing fault diagnosis based on the switchable normalization ssgan with 1-d representation of vibration signals as input," *Sensors*, vol. 19, no. 9, p. 2000, 2019.

[2] X. Li, W. Zhang, H. Ma, Z. Luo, and X. Li, "Data alignments in machinery remaining useful life prediction using deep adversarial neural networks," *Knowledge-Based Systems*, p. 105843, 2020.

[3] A. L. Ellefsen, E. Bjørlykhaug, V. Æsøy, S. Ushakov, and H. Zhang, "Remaining useful life predictions for turbofan engine degradation using semi-supervised deep architecture," *Reliability Engineering & System Safety*, vol. 183, pp. 240–251, 2019.

[4] G. S. Babu, P. Zhao, and X.-L. Li, "Deep convolutional neural network based regression approach for estimation of remaining useful life," in *International conference on database systems for advanced applications*. Springer, 2016, pp. 214–228.

[5] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long short-term memory network for remaining useful life estimation," in *2017 IEEE international conference on prognostics and health management (ICPHM)*. IEEE, 2017, pp. 88–95.

[6] X. Li, Q. Ding, and J.-Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering & System Safety*, vol. 172, pp. 1–11, 2018.

[7] C. Wu, F. Feng, S. Wu, P. Jiang, and J. Wang, "A method for constructing rolling bearing lifetime health indicator based on multi-scale convolutional neural networks," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 41, no. 11, p. 526, 2019.

[8] B. Yang, R. Liu, and E. Zio, "Remaining useful life prediction based on a double-convolutional neural network architecture," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9521–9530, 2019.

[9] X. Li, W. Zhang, and Q. Ding, "Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction," *Reliability Engineering & System Safety*, vol. 182, pp. 208–218, 2019.

[10] C. Cheng, G. Ma, Y. Zhang, M. Sun, F. Teng, H. Ding, and Y. Yuan, "Online bearing remaining useful life prediction based on a novel degradation indicator and convolutional neural networks," *arXiv preprint arXiv:1812.03315*, 2018.

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

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

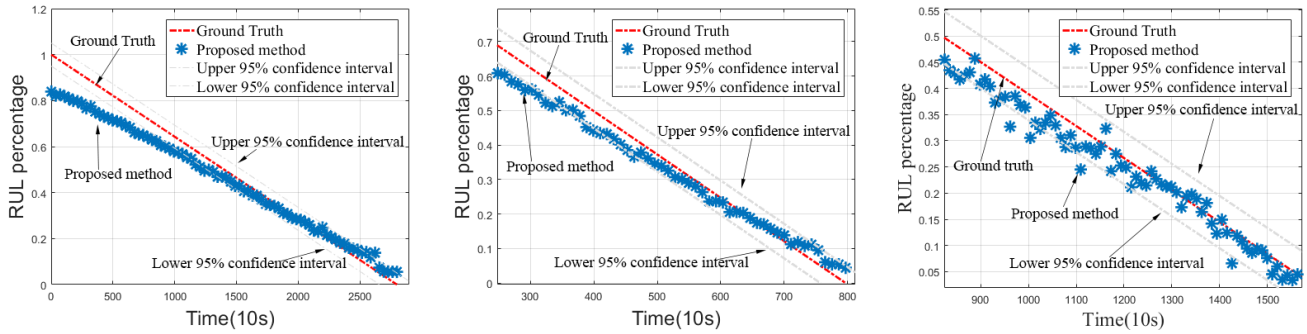
[13] A. Elsheikh, S. Yacout, and M.-S. Ouali, "Bidirectional handshaking lstm for remaining useful life prediction," *Neurocomputing*, vol. 323, pp. 148–156, 2019.

[14] J. Zhang, P. Wang, R. Yan, and R. X. Gao, "Long short-term memory for machine remaining life prediction," *Journal of manufacturing systems*, vol. 48, pp. 78–86, 2018.

[15] J. Chen, H. Jing, Y. Chang, and Q. Liu, "Gated recurrent unit based recurrent neural network for remaining useful life prediction of nonlinear deterioration process," *Reliability Engineering & System Safety*, vol. 185, pp. 372–382, 2019.

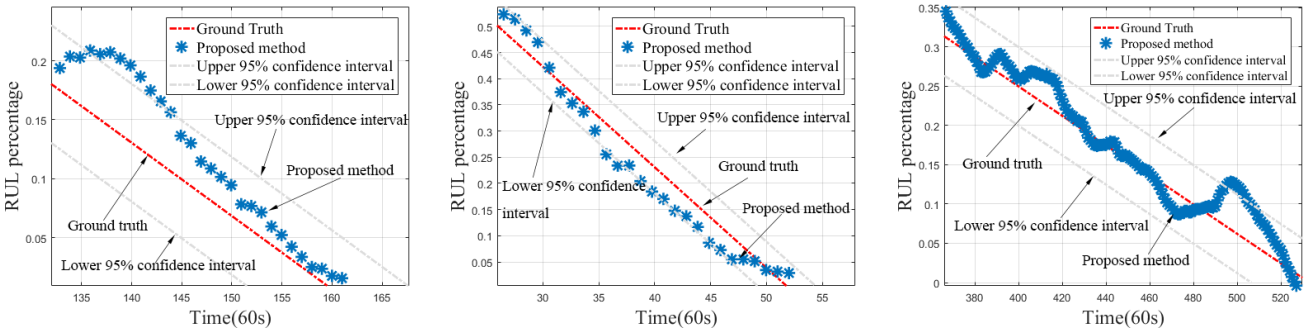
[16] L. Guo, Y. Lei, N. Li, T. Yan, and N. Li, "Machinery health indicator construction based on convolutional neural networks considering trend burr," *Neurocomputing*, vol. 292, pp. 142–150, 2018.

[17] Y. Wang, Y. Peng, Y. Zi, X. Jin, and K.-L. Tsui, "A two-stage data-driven-based prognostic approach for bearing degradation problem," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 3, pp. 924–932, 2016.



(a) A1-2. (b) A2-2. (c) A3-2.

Fig. 8: RUL Prediction on FEMTO Dataset.



(a) B1-2. (b) B1-5. (c) B2-3.

Fig. 9: RUL Prediction on XJTU-SY Dataset.

[18] Z. Gao, L. Qu, and G. Lu, "Early change detection in dynamic machine running status based on a new stability measure," *IEEE Transactions on Instrumentation and Measurement*, 2019.

[19] N. S. Keskar and R. Socher, "Improving generalization performance by switching from adam to sgd," *arXiv preprint arXiv:1712.07628*, 2017.

[20] FEMTO-ST, "Ieee phm 2012 data challenge," 2012.

[21] B. Wang, Y. Lei, N. Li, and N. Li, "A hybrid prognostics approach for estimating remaining useful life of rolling element bearings," *IEEE Transactions on Reliability*, 2018.