Machine Remaining Useful Life Prediction via an Attention-Based Deep Learning Approach

Zhenghua Chen, Min Wu, Rui Zhao, Feri Guretno, Ruqiang Yan, *Senior Member, IEEE*, and Xiaoli Li

***Abstract*—For prognostics and health management of mechanical systems, a core task is to predict the machine remaining useful life (RUL). Currently, deep structures with automatic feature learning, such as long short-term memory (LSTM), have achieved great performances for the RUL prediction. However, the conventional LSTM network only uses the learned features at last time step for regression or classification, which is not efficient. Besides, some handcrafted features with domain knowledge may convey additional information for the prediction of RUL. It is thus highly motivated to integrate both those handcrafted features and automatically learned features for the RUL prediction. In this article, we propose an attention-based deep learning framework for machine's RUL prediction. The LSTM network is employed to learn sequential features from raw sensory data. Meanwhile, the proposed attention mechanism is able to learn the importance of features and time steps, and assign larger weights to more important ones. Moreover, a feature fusion framework is developed to combine the handcrafted features with automatically learned features to boost the performance of the RUL prediction. Extensive experiments have been conducted on two real datasets and experimental results demonstrate that our proposed approach outperforms the state-of-the-arts.**

***Index Terms*—Attention mechanism, feature fusion, handcrafted features, long short-term memory (LSTM), machine remaining useful life (RUL) prediction, prognostics and health management (PHM).**

I. Introduction

REMAINING useful life (RUL) prediction is crucial for prognostics and health management (PHM) of mechanical systems. With an accurate RUL prediction, maintenance schedules can be designed to keep good working conditions for machines (or components), and thus, abrupt system failures can be avoided [1]. To achieve this objective, many advanced solutions have been developed, which generally can be divided into two categories, i.e., model based and data driven. For model-based solutions, they require to accurately model the dynamics of mechanical systems (or components) [2], [3]. However, due to the rapid development of industry, the mechanical systems become more and more complicated with complex interactions between each other. Therefore, accurate modeling of these systems is not realistic, even for experts. Besides, the flexibility and transferability of model-based solutions are poor, due to the distinct mechanisms for different mechanical systems.

Recently, data-driven solutions have attracted more and more attention for RUL prediction [4]. For data-driven solutions, it is not compulsory to know the detailed operation mechanism of mechanical systems. Instead, one only needs to collect some data from the systems, where the conditions of the systems can be identified based on data-driven algorithms. Generally, data-driven solutions can be further divided into statistic degradation modeling and artificial intelligence [4].

A typical statistic degradation modeling approach is the Cox's regression [5]. It models the hazard probability of an object based on the historical data about the life span of objects and their associate covariates. The hazard probability at time step t can be expressed as

$$\lambda(t|\mathbf{z}) = \lambda_0(t) \exp\left(\beta^\top \mathbf{z}\right) \tag{1}$$

where $\mathbf{z}$ is the covariates also known as features, $\lambda_0(t)$ is the baseline hazard function that changes over time, $\boldsymbol{\beta}$ is the regression coefficients, and $\top$ is the transpose operation. Many RUL prediction systems have been developed based on the Cox's regression. Pham *et al.* [6] presented an RUL prediction system that combines the Cox's proportional hazard model with a support vector machine (SVM) [6]. Liao *et al.* [7] compared the Cox's regression model with the logistic regression model for predicting the RUL of an individual unit [7]. You *et al.* [8] proposed a two-zone Cox's regression model for the equipment RUL prediction.

For artificial-intelligence-based RUL prediction, the objective is to directly build a relationship between the RUL of an object and the features that can be extracted manually from sensor measurements or automatically learned by deep learning algorithms. With the rapid development of artificial intelligence [9], in this work, we mainly focus on artificial-intelligence-based

Manuscript received May 7, 2019; revised September 30, 2019, November 28, 2019, and January 17, 2020; accepted January 24, 2020. Date of publication February 13, 2020; date of current version November 18, 2020. This work was supported in part by the A*STAR Industrial Internet of Things Research Program under the RIE2020 IAF-PP Grant A1788a0023, and in part by National Natural Science Foundation of China under Grant 51835009. *(Corresponding authors: Min Wu; Ruqiang Yan.)*

Zhenghua Chen, Min Wu, Feri Guretno, and Xiaoli Li are with the Institute for Infocomm Research, A*STAR, Sinagpore 138632, Singapore (e-mail: chen0832@e.ntu.edu.sg; wumin@i2r.a-star.edu.sg; guretnof@i2r.a-star.edu.sg; xlli@i2r.a-star.edu.sg).

Rui Zhao is with the Harveston Asset Management Company, Singapore 069542, Singapore (e-mail: rzhao001@e.ntu.edu.sg).

Ruqiang Yan is with the School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: yanruqiang@xjtu.edu.cn).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2020.2972443

0278-0046 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Nanyang Technological University. Downloaded on May 12,2021 at 10:28:40 UTC from IEEE Xplore. Restrictions apply.

RUL prediction, which includes the popular shallow machine learning and deep learning algorithms.

For shallow-model-based RUL prediction, it normally consists of feature extraction and inference. With domain knowledge on mechanical systems, some representative features can be extracted from raw sensory data that may be noisy and not representative for the RUL prediction [10], [11]. After extracting some informative features, the conventional machine learning algorithms, such as artificial neural network [12], [13], extreme learning machine (ELM) [14], SVM [15], [16], neural networks [17], and random forest (RF) [18], can be employed to predict the RUL.

Another popular algorithm for the artificial-intelligence-based RUL prediction is deep learning. Instead of manual feature extraction, deep learning is able to automatically learn representative features from raw sensory data [19], [20]. Besides, it can jointly optimize feature learning and RUL inference, leading to a better generalization performance for RUL prediction. For machine RUL prediction, most of sensory data are time series with temporal dependence. The deep learning approach of long short-term memory (LSTM), which is designed for sequential data analytics can be naturally suitable for the RUL prediction. A very good performance has been achieved for the RUL prediction using the LSTM approach [21], [22]. However, it still has several limitations for the conventional LSTM in the RUL prediction, which are presented as follows.

1) For the conventional LSTM network, it only uses the learned features at last time step for regression or classification. We argue that the learned features at other time steps may also have some contribution. And the learned features may have different contribution for the final RUL prediction. Therefore, an efficient operation is to assign larger weights to more important features and time steps.
2) Although the features learned by the LSTM network have been shown to be effective for the RUL prediction, some handcrafted features with domain knowledge also convey important information. The design of a network that can take both automatically learned features from the LSTM network and some meaningful handcrafted features into consideration for the final RUL prediction may boost the performance of RUL prediction.

To solve these limitations, we propose an attention-based deep learning framework for the RUL prediction. The proposed approach first exploits the LSTM network to learn representative sequential features from raw sensory data. Then, an attention network is developed to learn the importance of features and time steps, and assign larger weights to more important ones. Finally, we propose a feature fusion framework to make full use of all available information by combining the automatically learned features and some useful handcrafted features for the final prediction of the machine RUL. To verify the effectiveness of the proposed approach for the RUL prediction, we use real datasets for evaluation and compare it with various state-of-the-art methodologies for the machine RUL prediction. The main contributions of this article are summarized as follows.

1) We propose an attention-based deep learning framework for the machine RUL prediction. The attention network is able to learn the importance of features and time steps, and assign larger weights for more important ones to boost the performance of the RUL prediction.
2) Since some handcrafted features with domain knowledge may convey meaningful information for the RUL prediction, we propose a feature fusion framework to combine both automatically learned features and some meaningful handcrafted features for the final prediction of the RUL.
3) Real datasets are leveraged to evaluate the performance of the proposed approach. The results show that the proposed approach can significantly improve the performance of the RUL prediction.

II. Related Works

In this section, we review some advanced deep learning algorithms for machine's RUL prediction.

Deep learning is able to learn representative features and perform inference simultaneously, resulting remarkable performance for the RUL prediction. Babu *et al.* [23] proposed a deep convolutional neural network (CNN) for the RUL prediction [23]. The experimental results on two datasets indicated its superior performance for the RUL prediction when compared with some shallow learning algorithms. Zhu *et al.* [24] proposed a multiscale convolutional neural network (MSCNN) to predict the RUL [24]. First, a wavelet transform was conducted on raw sensory data to get time-frequency representation (TFR). Then, the TFR was used as the input of the MSCNN for the prediction of the RUL. Deutsch and He [25] presented a deep belief network feedforward neural network (DBN-FNN) for the RUL prediction [25]. The DBN was used to learn representative features, and the FNN was employed to perform the RUL prediction with the learned features. Zheng *et al.* [21] presented a long short-term memory (LSTM)-based RUL prediction. The experimental results on three datasets showed that the LSTM performs much better than some shallow learning algorithms and CNN [21]. Zhang *et al.* [22] proposed a bidirectional long short-term memory (BD-LSTM) approach to predict the RUL [22]. They first defined an health index (HI) based on a perceptron approach. The BD-LSTM was then utilized to track the variation of the HI for final RUL prediction. In [26], the authors proposed a multiobjective deep belief networks ensemble (MODBNE) approach for the prediction of the RUL. They applied a multiobjective evolutionary algorithm to train DBNs with two conflict objectives, i.e., accuracy and diversity. The evolved DBNs were then combined to form an ensemble model for the final RUL prediction.

Since the sensory data for PHM are time series with temporal dependence, the LSTM network that performs well for sequential data modeling is naturally suitable for the RUL prediction. However, different features at different time steps learned by the LSTM network will have equal contribution for the final prediction of the RUL, which is not effective. A more effective operation is to assign larger weights for more important features and time steps. Therefore, in this article, we propose an attention mechanism with the LSTM to automatically assign larger weights to more significant features and time steps

Authorized licensed use limited to: Nanyang Technological University. Downloaded on May 12,2021 at 10:28:40 UTC from IEEE Xplore. Restrictions apply.

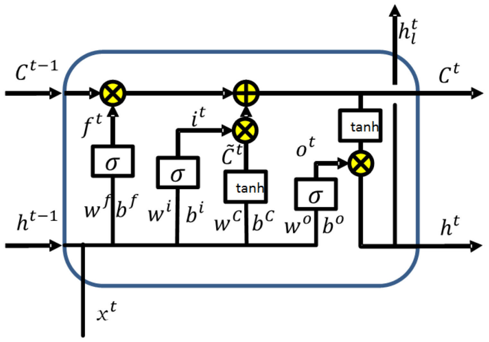


Fig. 1. Structure of the LSTM.

to boost the performance of the RUL prediction. Meanwhile, some handcrafted features may convey useful information for the prediction of the RUL. Hence, we develop a feature fusion framework to combine these handcrafted features with the features learned by the attention-based LSTM to further improve the performance of the machine RUL prediction.

III. Methodology

A. Long Short-Term Memory (LSTM)

To predict the RUL of machines, a number of sensors, such as vibration, temperature, acoustic, etc., should be deployed. Generally, the sensor measurements are time series with temporal dependence. Recurrent neural network (RNN) whose nodes are connected along a sequence was designed to model temporal dependence in time series [27]. Therefore, an RNN is naturally suitable for machine RUL prediction leveraging sequential sensor measurements. However, the conventional RNN often suffers from the problem of gradient vanishing or exploding during the network training, which greatly degrades its performance on modeling long-term dependencies [28]. To solve this problem, Hochreiter and Schmidhuber proposed a new architecture, named LSTM, which can be treated as a memory cell that consists of a few gates [29]. The gates, which can allow or prevent the passing of information along a sequence, can capture long-term dependencies. Owing to its unique property, the LSTM network has achieved great successes in the analysis of time-series data, such as occupancy estimation [30], video analysis [31], and nature language processing [32]. Recently, it also has achieved great performance for the RUL prediction [21], [22], [33].

A typical LSTM network is shown in Fig. 1. It consists of a forget gate to discard the unnecessary information from previous time steps, an input gate to select useful information from inputs, and an output gate to control the outputs of the current LSTM network. Assuming that x^t is the input at time step t, h^t is the hidden state at time step t, C^{t-1} is the memory cell state, w^f, w^i, w^C, and w^o are the weights, b^f, b^i, b^C, and b^o are the biases, and $\sigma(\cdot)$ and tanh are the *sigmoid* and tanh functions, respectively, the LSTM network can be expressed as

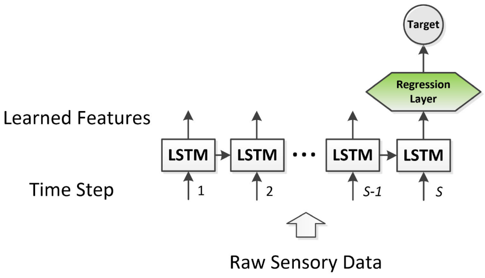


Fig. 2. Standard LSTM for regression problems.

$$\begin{aligned}
f^t &= \sigma\left(w^f[h^{t-1}, x^t] + b^f\right)\\
i^t &= \sigma\left(w^i[h^{t-1}, x^t] + b^i\right)\\
\tilde{C}^t &= \tanh\left(w^C[h^{t-1}, x^t] + b^C\right)\\
C^t &= f^t * C^{t-1} + i^t * \tilde{C}^t\\
o^t &= \sigma\left(w^o[h^{t-1}, x^t] + b^o\right)\\
h^t &= o^t * \tanh\left(C^t\right).
\end{aligned} \tag{2}$$

Due to the strong sequential modeling ability of the LSTM network, it has been successfully used for machine RUL prediction in [21] and [22]. However, they applied the standard LSTM that only uses the learned features at last time step for regression, which is shown in Fig. 2. We argue that the learned features at other time steps may also have some contribution. And the learned features may have different contribution for the final RUL prediction. Therefore, we intend to design an attention mechanism to learn the importance of features and time steps. The details will be introduced in the following paragraphs.

B. Attention Mechanism

The attention mechanism is first proposed for the task of image processing, inspired by human vision system [34], [35]. Human always pays attention to a certain region of an image during recognition, indicating that different weights will be assigned to different regions of an image. The attention mechanism has been successfully applied for a number of applications, such as language translation [36] and time-series prediction [37].

For the task of machine RUL prediction, an efficient operation is to focus on different region of interest by assigning different weights for different features at different time steps. In this task, since no prior information is available, we leverage a self-attention mechanism to learn the importance of features and time steps. Assume that the learned features by the LSTM network for one sample can be expressed as $\mathbf{H} = \{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_d\}^\top$, $\top$ is the transpose operation. Here, $\mathbf{h}_i \in \mathbb{R}^n$, where n is the number of sequential steps of the features. Based on the self-attention mechanism, the importance for different sequential steps of ith input $\mathbf{h}_i$ can be expressed as

$$\mathbf{s}_i = \Phi(\mathbf{W}^\top \mathbf{h}_i + \mathbf{b}) \tag{3}$$

where $\mathbf{W}$ and b are the weight matrix and the bias vector, respectively, $\Phi(\cdot)$ is the score function that can be designed as an activation function in neural networks, such as *sigmoid* and *linear*. After obtaining the score for the ith feature vector, it can

Authorized licensed use limited to: Nanyang Technological University. Downloaded on May 12,2021 at 10:28:40 UTC from IEEE Xplore. Restrictions apply.

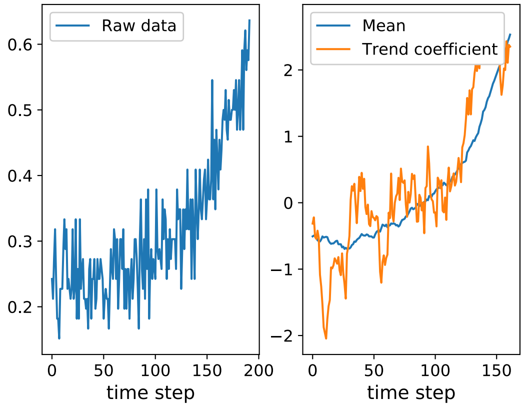


Fig. 3. Example for the features of mean and trend coefficient.

be normalized using a *softmax* function as follows:

$$\mathbf{a}_i = softmax(\mathbf{s}_i) = \frac{\exp(\mathbf{s}_i)}{\sum_i \exp(\mathbf{s}_i)}. \tag{4}$$

The final output feature $\mathbf{O}$ of the attention mechanism can be expressed as

$$\mathbf{O} = \mathbf{H} \otimes \mathbf{A} \tag{5}$$

where $\mathbf{A} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_d\}$, and $\otimes$ is a new operation defined as element-wise multiplication. Given vectors $\mathbf{b} = [b_1 \quad b_2 \quad \ldots \quad b_n]^\top$ and $\mathbf{c} = [c_1 \quad c_2 \quad \ldots \quad c_n]^\top$, $\mathbf{b} \otimes \mathbf{c} = [b_1c_1 \quad b_2c_2 \quad \ldots \quad b_nc_n]^\top$.

C. Attention-Based Deep Learning for RUL Prediction

1) Handcrafted Features: For sensory-data-based RUL prediction, some intuitive handcrafted features can be extracted, such as mean and trend coefficient of linear regression. The mean value shows the magnitude of sensory data, and the trend coefficient indicates the degradation of sensory data. These two simple handcrafted features have been shown to be effective for RUL prediction in [16]. An example of these two features is shown in Fig. 3. Note that, the features have been standardized for normalization. It can be found that these two features well indicate the properties of the raw sensory data.

To make full use of all the available information, we propose a feature fusion framework to combine the features learned by deep structures with some meaningful handcrafted features to boost the performance of the RUL prediction. The details of the proposed approach will be shown in the following paragraphs.

2) Proposed Framework: Fig. 4 shows the proposed attention-based deep learning framework for the prediction of machine RUL. First, the raw sensory data are fed into the LSTM network for feature learning. The learned sequential features are treated as the inputs of the attention model, whose outputs (attention weights) indicate the importance of features and time steps. Then, the learned sequential features are merged with the weights generated by the attention model. After that, two fully connected layers (FCLs) are performed to obtain more abstract features. Meanwhile, the handcrafted features are extracted from the raw sensory data, and then fed into an FCL to obtain more abstract features. To make full use of these two types of features, we concatenate them to form a complete feature set. Finally, a regression layer is used for the RUL prediction. Table I summarizes the inputs of some key modules, including the LSTM network, the attention layer, the merge layer, and the fully connected layer (i.e., the FCL at the right-hand side in Fig. 4).

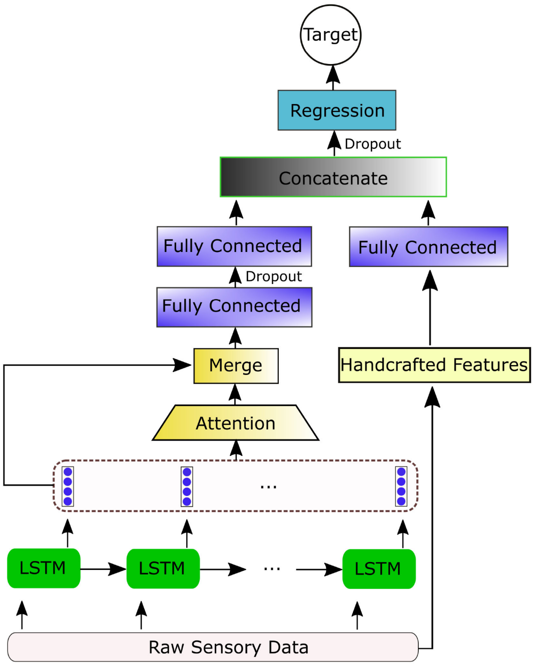


Fig. 4. Proposed attention-based deep learning approach for the RUL prediction.

TABLE I
INPUTS OF SOME KEY MODULES

Module	Input
LSTM network	Raw sensory data
Attention layer	Features learned by the LSTM
Merge layer	Features learned by the LSTM; Weights generated by the attention layer
Fully connected layer (Right-hand side in Fig. 4)	Handcrafted features

Since the prediction of the RUL is a typical regression problem, the loss function of the proposed approach is set to be the mean square error (MSE) loss, which is defined in Appendix. Given the predicted RULs and the true RULs, the MSE losses over training data can be calculated and back-propagated to generate the error gradients for each layer (such as an LSTM layer, attention layer, and fully connected layers). Then, the optimization method of *Adam* that computes adaptive learning rates for each parameter [38] is adopted to optimize model parameters at each layer based on the error gradients. Note that, the proposed framework is an end-to-end trainable architecture, which means that all the model parameters can be jointly trained. Considering the overfitting problem in deep learning models, appropriate regularization techniques should be performed. Dropout is one of the most popular techniques for solving this problem [39]. In dropout, parts of the hidden outputs are randomly masked so that these neurons will not influence the forward propagation during the model training. When it comes to testing, the dropout will

Authorized licensed use limited to: Nanyang Technological University. Downloaded on May 12,2021 at 10:28:40 UTC from IEEE Xplore. Restrictions apply.

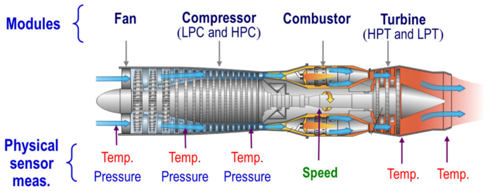


Fig. 5. Diagram of aircraft engine [40].

TABLE II
DESCRIPTION OF THE DATASETS FOR EVALUATION [40], [41]

Dataset	C-MAPSS		PHM 2008
	FD001	FD004	
No. of training engines	100	249	218
No. of testing engines	100	248	218
Working conditions	1	6	6
Faulty types	1	2	2

be turned OFF and the outputs of all hidden neurons will make effects. In other view, it can be treated as a kind of ensemble learning. In this work, two dropout layers are adopted (see Fig. 4). The masking probability for dropout is set to be 0.5.

IV. EVALUATION

A. Data Description

The widely used commercial modular aero-propulsion system simulation (C-MAPSS) dataset [40] is adopted for the evaluation of the proposed approach. This dataset describes the degradation process of the aircraft engine whose diagram is shown in Fig. 5. The engine consists of fan, low pressure compressor (LPC), high pressure compressor (HPC), combustor, low pressure turbine (LPT), and high pressure turbine (HPT). Total 21 onboard sensors, measuring temperature, pressure, and speed, are deployed at different locations to monitor the condition of the engine.

The entire dataset contains four subdatasets with varying number of operation conditions and faulty types. We choose two typical subdatasets: the simplest one FD001 that contains one operation condition and one faulty type and the most complicated one FD004 that contains six operation conditions and two faulty types. For FD001 and FD004, they both contain two files for training and testing. The training file records sensor data at each running cycle in the run-to-fail experiments for certain number of engines. The testing file only contains the sensor measurements to certain running cycles for another certain number of engines. The objective is to predict the RUL of each engine in the testing file with the given sensor measurements.

Another widely used dataset for evaluation is the PHM 2008 dataset [41], which has the same data structure with the C-MAPSS dataset, but different number of training and testing engines. The description of the datasets is shown in Table II.

For the 21 sensors (indices from 1 to 21 in training and testing files), the sensors with indices 1, 5, 6, 10, 16, 18, and 19 always have constant values during the run-to-fail experiments. This means that these sensors are not related to the degradation of engines. Hence, these sensors are removed from the two datasets as did in [26] and [42]. Finally, 14 sensors are used for the RUL prediction. Since different operating conditions will influence the RUL, we treat operating conditions as measurement signals for the RUL prediction. Therefore, the operating conditions and sensor measurements are the final inputs of prediction models.

B. Data Preprocessing

A sliding window is commonly used for data segmentation [23], [26]. An example of data segmentation for training samples is shown in Fig. 6. For the run-to-fail experiments, assume that the number of total running cycles of an engine is T, the window size is s and the step size is p. Each sample will have a size of $s \times n$, where n is the number of sensors. According to Fig. 6, the RUL of the $(i+1)$th sample (window) is $T - s - i * p$. Note that, a piece-wise linear RUL [23], [26] is used instead of the true RUL for training data, which means that if the true RUL is larger than the maximal RUL, it will be set to the maximal RUL. Here, we choose a window size of 30 and a step size of 1, which are the same as these in [26]. Under these settings, the numbers of training samples for FD001, FD004, and PHM 2008 are 17 731, 54 028, and 39 596, respectively. For testing, only one data window to the last sensor measurements for each engine is used as the testing sample. Hence, the numbers of testing samples for FD001, FD004, and PHM 2008 are 100, 248, and 218, respectively, which are the numbers of testing engines in the three datasets. Note that, the actual RUL values of testing samples for FD001 and FD004 are available to the public, while the actual RUL values of testing samples for PHM 2008 are not available.

C. Evaluation Criteria

To evaluate the performance of the RUL prediction, two widely used evaluation criteria, i.e., root mean square error (RMSE) and scoring function, are adopted [23], [26]. The definition of the RMSE can be found in Appendix. In machine RUL prediction, the late prediction refers to that the predicted RUL is larger than the actual RUL. Late predictions will generally lead to more severe consequences than the early prediction, however, this fact cannot be reflected by the criterion of the RMSE. Thus, the scoring function [23] (see the Appendix for the definition) can be utilized. Based on the definition, more penalties will be given to late predictions, which is consistent with our common sense. Both criteria are useful to evaluate the performance of the RUL prediction. Fig. 7 compares the RMSE with the scoring function. Both criteria are useful to evaluate the performance of the RUL prediction. Our experimental results will be reported based on these two criteria.

D. Experimental Setup

To verify the effectiveness of the proposed approach, we first perform an initial test on the training data of FD001, which is split for training and testing. A comparison has been made between the proposed approach and some widely used benchmark approaches including SVR, RF, CNN, and LSTM. We have also compared the proposed approach with some state-of-the-art approaches on the testing data of FD001, FD004, and PHM 2008

Authorized licensed use limited to: Nanyang Technological University. Downloaded on May 12,2021 at 10:28:40 UTC from IEEE Xplore. Restrictions apply.

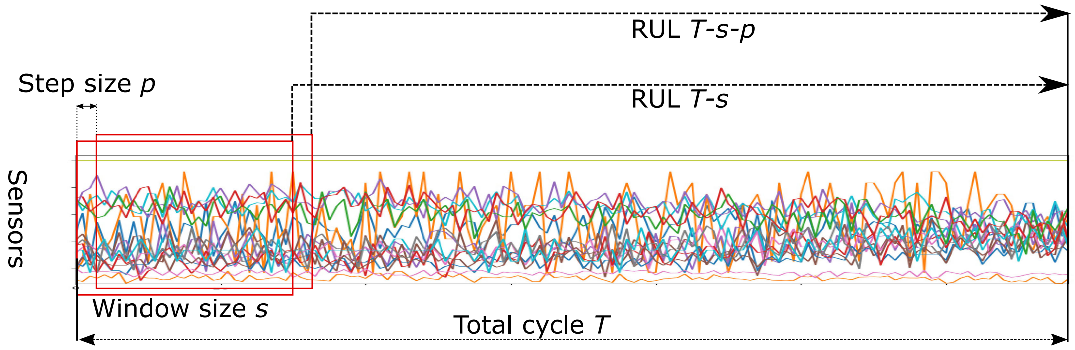


Fig. 6. Example of data segmentation for the RUL prediction.

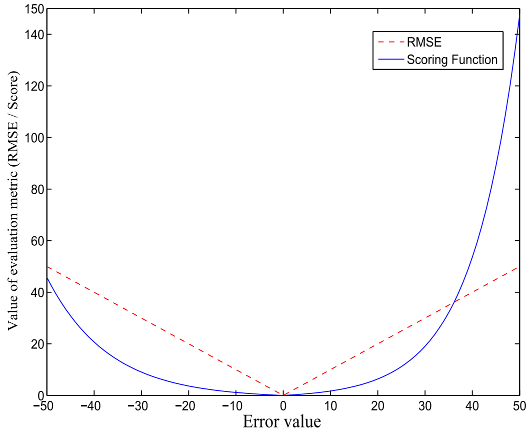


Fig. 7. RMSE versus scoring function [23].

datasets. Note that, the state-of-the-arts used the same data for training and testing.

For the proposed attention-based deep learning approach, some structural parameters, i.e., the number of hidden nodes, should be tuned based on the given training data. A cross validation is performed with the training data to determine the parameters of the proposed approach. Specifically, the number of hidden units for the LSTM network is 50. The hidden nodes of two FCLs after the merge layer are set to be [50, 10]. And the hidden nodes of the FCL on the handcrafted features are set to be 10. The learning rate for the optimization algorithm of *Adam* is set to be 0.001. For the algorithms of SVR, RF, and CNN, we use the parameters in [23] and [26], which have carefully tuned the parameters of the models. For the algorithm of the conventional LSTM, the parameters have been carefully tuned by using cross validation on the training data. Specifically, the number of hidden nodes for the LSTM network is set to be 50. Two FCLs with the sizes of [50, 10] are applied for the RUL prediction. Considering the randomness in parameter initialization of some algorithms, i.e., CNN, LSTM, and the proposed approach, these algorithms are run ten times for each dataset and average results are reported. The source code of the proposed approach is available at the open-source GitHub.[1]

E. Initial Test on FD001

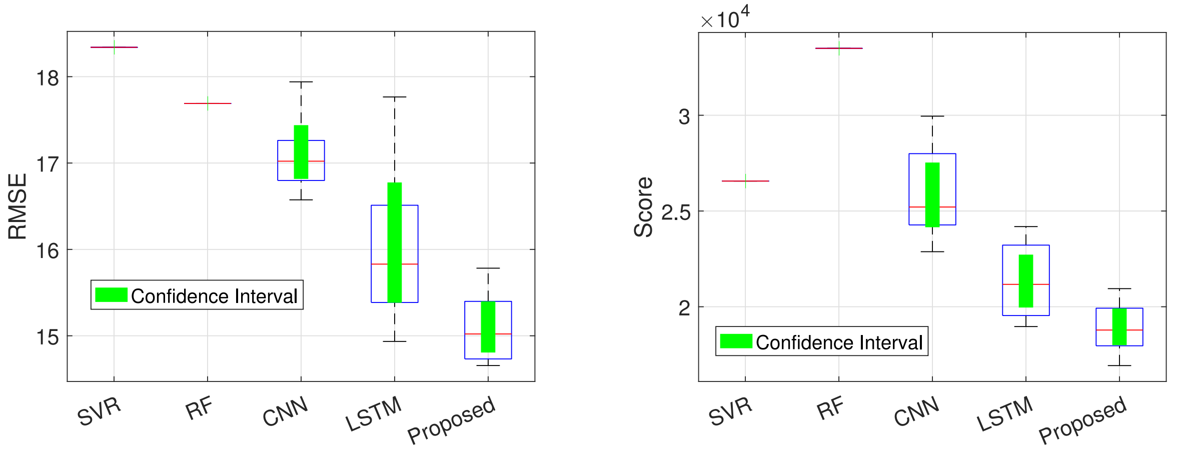
In order to have more samples for testing, an initial test is first performed with the training data in FD001. By using sliding windows, 17 731 samples can be obtained. Here, we use first 70% of data (12 412 samples) for model training and the rest (5 319 samples) for testing. A comparison has been made with some widely used benchmark algorithms for the RUL prediction, including SVR, RF, CNN, and LSTM. The results are shown in Fig. 8. We also indicate the 95% confidence interval of all the results in the figure. It can be found that, due to the powerful feature learning ability of deep structures, deep-learning-based algorithms perform better than show models, i.e., SVR and RF. Due to the sequential modeling ability of the LSTM, it has a superior performance over the CNN. Owing to the proposed attention mechanism and the feature fusion framework, the proposed approach outperforms all the benchmark approaches.

We also present the training and testing time for different algorithms on a workstation that has 12 core CPUs of Intel i7-8700 3.20 GHz and a GPU of NVIDIA GeForce GTX1080Ti. The results are shown in Table III. Obviously, deep-learning-based approaches require much more training and testing time than shallow models, because of much more parameters to be optimized. The proposed approach has the longest training time, which is 110.15 s. Considering that the model training only requires to be done once and it is offline, this amount of time for training is still acceptable. The testing time of the proposed method for all the 5319 samples is only 0.42 s, which means that the testing time for one sample is 7.8×10^{-5} that can be neglected for most of industrial applications. Hence, it can be claimed that the proposed method can be used for real-time implementations.

To better interpret the attention mechanism for the RUL prediction, an attention matrix of one testing sample is illustrated in Fig. 9. Since we set 50 hidden nodes for the LSTM network and the sliding window contains 30 time steps, the final learned features by the LSTM will have a dimension of 30 × 50. In the conventional LSTM without the attention mechanism, only the learned features at last time step of a sample will be used for classification or regression problems. We argue that the learned features at other time steps may also have some contribution for

[1] [Online]. Available: https://github.com/ZhenghuaNTU/RUL-prediction-using-attention-based-deep-learning-approach

Authorized licensed use limited to: Nanyang Technological University. Downloaded on May 12,2021 at 10:28:40 UTC from IEEE Xplore. Restrictions apply.



(a)

(b)

Fig. 8. Experimental results of all the benchmark approaches and the proposed approach in the initial test on FD001. (a) RMSE. (b) Score.

TABLE III
TRAINING AND TESTING TIME OF THE BENCHMARK APPROACHES AND THE PROPOSED APPROACH

	SVR	RF	CNN	LSTM	Proposed
Training time (s)	2.43	6.80	27.77	81.75	110.15
Testing time (s)	0.45	0.010	0.11	0.31	0.42

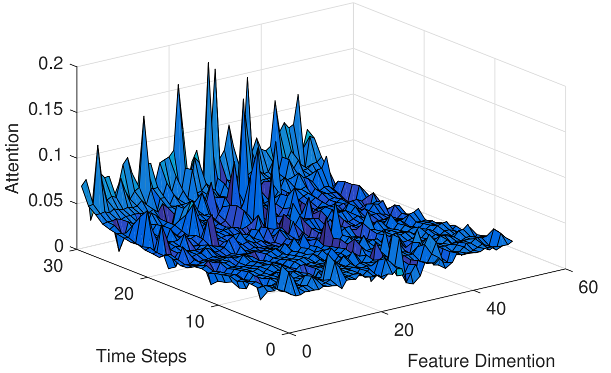
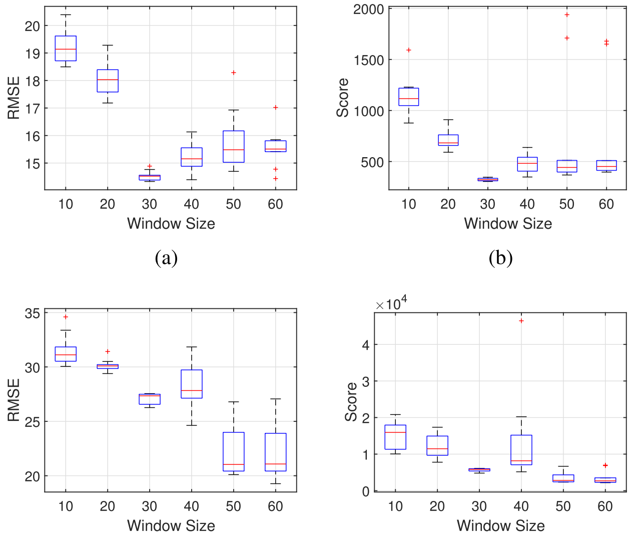


Fig. 9. Attention matrix of one sample.



(a)

(b)

(c)

(d)

Fig. 10. Experimental results of the proposed approach with different window sizes on the two datasets. (a) FD001: RMSE. (b) FD001: Score. (c) FD004: RMSE. (d) FD004: Score.

the RUL prediction. From Fig. 9, it can be found that more recent time steps will have larger attention weights, indicating that more recent steps will be more important for the RUL prediction. This is consistent with our common sense. Although the learned features at each time step cannot be explicitly interpreted because they are high-level features learned by the LSTM network, we can still concur that they will have different contribution for the final RUL prediction, which has been achieved by using the attention mechanism.

F. Results on the Testing Data of FD001 and FD004

Here, we first analyze the impact of the window size on the performance of the RUL prediction. Then, an ablation study is performed to demonstrate the effectiveness of the proposed attention mechanism and feature fusion. Finally, we compare the proposed approach with some state-of-the-art methods for the RUL prediction that use the same data for model training and testing.

1) Analysis on Window Size: For data preprocessing, window size is one of the most important parameters. To evaluate the impact of this parameter, we implement the proposed approach with different window sizes, i.e., 10, 20, 30, 40, 50, and 60, on the two datasets. The results are shown in Fig. 10. For the simple FD001 dataset, it can be found that the performance is enhanced when increasing window size at the beginning. This is because more information is included for the RUL prediction. However, after increasing the window size over a certain value, i.e., 30, the performance of model degrades on this dataset. This may be caused by the overfitting of the algorithm with too much

Authorized licensed use limited to: Nanyang Technological University. Downloaded on May 12,2021 at 10:28:40 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
RESULTS OF THE ABLATION STUDY

Methods	FD001		FD004	
	RMSE	Score	RMSE	Score
LSTM + 2 FCs + Regression (Ori-LSTM)	15.43	410.60	29.12	12551.44
LSTM + Attention + 2 FCs + Regression (LSTM-Attention)	15.27	362.92	28.04	7850.40
[LSTM + 2 FCs, Handcrafted Feat + FC] + Regression (LSTM+HF)	14.81	360.72	27.66	7812.37
[LSTM + Attention + 2 FCs, Handcrafted Feat + FC] + Regression (Proposed)	14.54	322.44	27.08	5649.14

TABLE V
EXPERIMENTAL RESULTS ON THE FD001 AND FD004 DATASETS

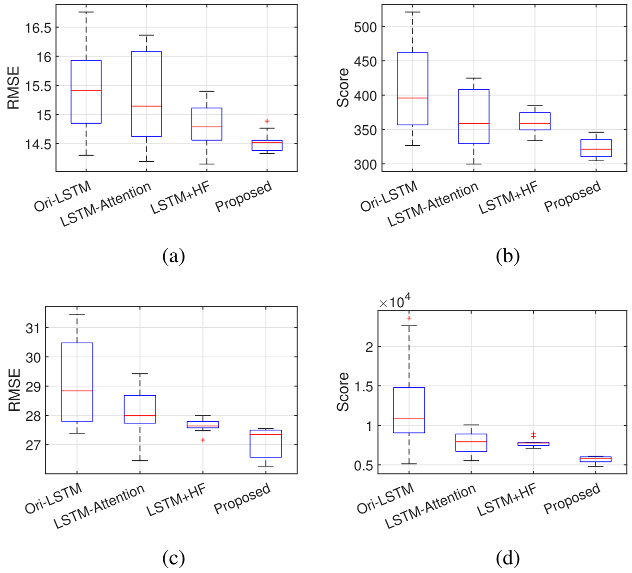
	Criterion	Statistic	Shallow learning					Deep learning				Hybrid
		Cox's regression	MLP [23]	SVR [23]	RVR [23]	ELM [26]	RF [26]	CNN [23]	LSTM [21], [22]	DBN [26]	MODBNE [26]	Proposed
FD001	RMSE	45.10	37.56	20.96	23.80	17.27	17.91	18.45	15.42	15.21	15.04	**14.53**
	Score	28616	17972	1381.5	1502.9	523.00	479.75	1286.7	410.60	417.59	334.23	**322.44**
FD004	RMSE	54.29	77.37	45.35	34.34	38.43	31.12	29.16	29.12	29.88	28.66	**27.08**
	Score	1164590	5616600	371140	26509	121414.47	46567.63	7886.4	12551.44	7954.51	6557.62	**5649.14**

information for this simple dataset. For the much more complicated FD004 dataset, similarly, the performance of the model is enhanced when the number of hidden nodes increases from 10 to 30. After 30, the performance slightly degrades, which may be because the other model parameters are tuned under 30, resulting a slight degradation of the performance with different number of hidden nodes. Moreover, we interestingly find that when we further increase the number of hidden nodes, the performance of the model improves a lot on this complicated dataset. This means that the complicated dataset may require more information for an accurate RUL prediction. The performance of the model under the window sizes of 50 and 60 is similar, which should be caused by the limited modeling capacity of the model with large amount of information. Therefore, we may be able to further enhance the performance of the model by using more hidden nodes or deeper structure.

In summary, it can be concluded that more complicated datasets require a larger window size to include more information for an accurate RUL prediction. In real applications, we should choose different window sizes for different datasets by using cross validation on the training data. To give a fair comparison with state-of-the-arts that used a sliding window of 30 in their works [22], [26], a window size of 30 is chosen in this work.

2) Ablation Study of the Proposed Approach: To evaluate the effectiveness of the proposed attention mechanism and feature fusion, an ablation study of the proposed approach is performed. Specifically, we implement the original LSTM, the LSTM with attention, the LSTM plus handcrafted features and the proposed framework. The results can be found in Table IV and Fig. 11.

Generally, the LSTM with attention and the LSTM plus feature fusion outperform the original LSTM, which indicates the effectiveness of the two proposed schemes. Moreover, it can be found that the LSTM plus feature fusion has a superior performance than the LSTM with attention mechanism in terms of the RMSE, Score, and their variances. This means that the proposed feature fusion is more effective than the proposed attention mechanism to enhance the performance of the original LSTM for the RUL prediction. In other words, the handcrafted features well compensate the automatically learned features for the task of the RUL prediction. The proposed approach with these two effective schemes achieves the best performance on the two datasets in terms of error and variance under the two criteria.

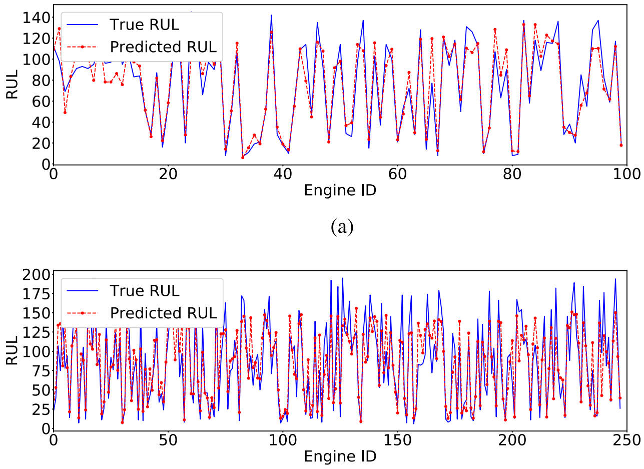


(a) (b) (c) (d)

Fig. 11. Experimental results of the ablation study. (a) FD001: RMSE. (b) FD001: Score. (c) FD004: RMSE. (d) FD004: Score.

3) Compare to State-of-the-Arts: The experimental results of the proposed approach and some state-of-the-art approaches on the two datasets are shown in Table V. Overall, all the approaches perform better on the FD001 than that on the FD004. This is because the FD001 is relatively simple with only one operation condition and one faulty type. Besides, the number of engines for testing in the FD004 is 248, which is much larger than that in the FD001. Therefore, the scores that are summations over all the engines of the FD004 and the FD001 are under different magnitude.

Authorized licensed use limited to: Nanyang Technological University. Downloaded on May 12,2021 at 10:28:40 UTC from IEEE Xplore. Restrictions apply.



(a)

(b)

Fig. 12. True RUL and the predicted RUL by the proposed approach on the two datasets. (a) FD001. (b) FD004.

According to Table V, deep-learning-based methods, such as LSTM, DBN, and MODBNE, perform better than statistic learning algorithm of Cox's regression and shallow learning algorithms, i.e., MLP, SVR, RVR, ELM, and RF. This indicates the powerful feature learning ability of deep structures. The MODBNE, which is an ensemble of the DBN, outperforms the other benchmark approaches, due to the feature learning of the DBN and the ensemble structure. However, the ensemble structure of the DBN will have a much higher computational complexity than other shallow and deep structures. Owing to the proposed attention mechanism and feature fusion framework, the proposed approach outperforms all these state-of-the-art approaches, including the powerful ensemble deep learning approach of MODBNE.

The predicted RUL on the two datasets is shown in Fig. 12. For both datasets, the predicted RUL matches very well with the true RUL, which indicates the feasibility of the RUL prediction. Since FD001 is relatively simple, the prediction performance on FD001 is better than that on FD004, which is consistent with our previous analysis based on Table V.

G. Results on the PHM 2008 Dataset

For the PHM 2008 dataset, the actual RUL values of testing samples are not available. The results need to be uploaded to the NASA Data Repository Website[2] where a score value [see (8)] will be given.

The results on the PHM 2008 dataset are shown in Table VI. It can be found that deep-learning-based solutions performs much better than statistic and shallow learning algorithms. Due to the sequential modeling capacity of the LSTM, it outperforms the popular CNN approach. The proposed approach has a superior performance over all the benchmark approaches, which indicates its effectiveness on the RUL prediction.

TABLE VI
Experimental Results on the PHM 2008 Dataset

Method	Score
Cox's regression	65558
MLP [23]	3212
SVR [23]	15886
RVR [23]	8242
CNN [23]	2056
LSTM [21], [22]	1862
Proposed	**1584**

V. Conclusion

In this article, we proposed an attention-based deep learning framework for the machine RUL prediction. First, we employed the deep learning algorithm of the LSTM for automatic feature learning from raw sensory data. Then, an attention mechanism was proposed to learn the importance of features and time steps, and automatically assign larger weights to more important ones. Meanwhile, some handcrafted features with domain knowledge may convey additional information for the RUL prediction. Hence, a feature fusion framework was designed to combine the handcrafted features and the automatically learned features to boost the performance of the RUL prediction. The proposed approach was evaluated using real datasets. Since the size of the sliding window is important for the RUL prediction, the impact of different window size on the prediction performance was investigated. Then, we verified the effectiveness of the proposed attention mechanism and feature fusion for the machine RUL prediction. Finally, a comparison was made with various state-of-the-art approaches. The proposed approach outperforms these state-of-the-arts under two popular evaluation criteria.

For machine-learning-based RUL prediction, the basic assumption is that the underline patterns between training and testing data are the same. However, if the training and testing data are collected under different environments, working conditions, or machines, the underline patterns between training and testing data may be distinct, which will hinder the performance of machine-learning-based methods [43]. To solve this issue, transfer learning, which is able to transfer the knowledge learned from one domain to another [44] can be adopted. In our future works, we will investigate transfer-learning-based RUL prediction with varying environments, working conditions, orenlrg machines.

Appendix

Loss Function: The MSE loss is defined as follows:

$$\text{MSE Loss} = \frac{1}{N}\sum_{i=1}^{N}(\hat{r}_i - r_i)^2 \tag{6}$$

where r_i and $\hat{r}_i$ are the true RUL and the predicted RUL, respectively, and N is the total number of samples.

Evaluation Metrics: The definition of the RMSE is as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum\nolimits_{i=1}^{N}(\hat{r}_i - r_i)^2} \tag{7}$$

[2] [Online]. Available: https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/

Authorized licensed use limited to: Nanyang Technological University. Downloaded on May 12,2021 at 10:28:40 UTC from IEEE Xplore. Restrictions apply.

where r_i and $\hat{r}_i$ are the true RUL and the predicted RUL, respectively. The scoring function [23] is defined as

$$S = \begin{cases} \sum_{i=1}^{N} (e^{-\frac{\hat{r}_i - r_i}{13}} - 1), & \text{when } \hat{r}_i < r_i \\ \sum_{i=1}^{N} (e^{\frac{\hat{r}_i - r_i}{10}} - 1), & \text{when } \hat{r}_i \geq r_i \end{cases}. \tag{8}$$

References

[1] L. Liao, "Discovering prognostic features using genetic programming in remaining useful life prediction," *IEEE Trans. Ind. Electron.*, vol. 61, no. 5, pp. 2464–2472, May 2014.

[2] Y. Qian, R. Yan, and R. X. Gao, "A multi-time scale approach to remaining useful life prediction in rolling bearing," *Mech. Syst. Signal Process.*, vol. 83, pp. 549–567, 2017.

[3] Q. Zhai and Z.-S. Ye, "RUL prediction of deteriorating products using an adaptive wiener process model," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 2911–2921, Dec. 2017.

[4] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, "Machinery health prognostics: A systematic review from data acquisition to RUL prediction," *Mech. Syst. Signal Process.*, vol. 104, pp. 799–834, 2018.

[5] C. R. David *et al.*, "Regression models and life tables (with discussion)," *J. Roy. Statistical Soc.*, vol. 34, no. 2, pp. 187–220, 1972.

[6] H. T. Pham *et al.*, "Machine performance degradation assessment and remaining useful life prediction using proportional hazard model and support vector machine," *Mech. Syst. Signal Process.*, vol. 32, pp. 320–330, 2012.

[7] H. Liao, W. Zhao, and H. Guo, "Predicting remaining useful life of an individual unit using proportional hazards model and logistic regression model," in *Proc. Annu. Rel. Maintainability Symp.*, 2006, pp. 127–132.

[8] M.-Y. You, L. Li, G. Meng, and J. Ni, "Two-zone proportional hazard model for equipment remaining useful life prediction," *J. Manuf. Sci. Eng.*, vol. 132, no. 4, 2010, Art. no. 041008.

[9] X.-S. Si, W. Wang, C.-H. Hu, and D.-H. Zhou, "Remaining useful life estimation—A review on the statistical data driven approaches," *Eur. J. Oper. Res.*, vol. 213, no. 1, pp. 1–14, 2011.

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

[11] T. Benkedjouh, K. Medjaher, N. Zerhouni, and S. Rechak, "Remaining useful life estimation based on nonlinear feature reduction and support vector regression," *Eng. Appl. Artif. Intell.*, vol. 26, no. 7, pp. 1751–1760, 2013.

[12] N. Gebraeel, M. Lawley, R. Liu, and V. Parmeshwaran, "Residual life predictions from vibration-based degradation signals: A neural network approach," *IEEE Trans. Ind. Electron.*, vol. 51, no. 3, pp. 694–700, Jun. 2004.

[13] Z. Tian, "An artificial neural network method for remaining useful life prediction of equipment subject to condition monitoring," *J. Intell. Manuf.*, vol. 23, no. 2, pp. 227–237, 2012.

[14] K. Javed, R. Gouriveau, and N. Zerhouni, "A new multivariate approach for prognostics based on extreme learning machine and fuzzy clustering," *IEEE Trans. Cybern.*, vol. 45, no. 12, pp. 2626–2639, Dec. 2015.

[15] P. G. Nieto, E. Garcia-Gonzalo, F. S. Lasheras, and F. J. de Cos Juez, "Hybrid PSO–SVM-based method for forecasting of the remaining useful life for aircraft engines and evaluation of its reliability," *Rel. Eng. Syst. Safety*, vol. 138, pp. 219–231, 2015.

[16] R. Khelif, B. Chebel-Morello, S. Malinowski, E. Laajili, F. Fnaiech, and N. Zerhouni, "Direct remaining useful life estimation based on support vector regression," *IEEE Trans. Ind. Electron.*, vol. 64, no. 3, pp. 2276–2285, Mar. 2017.

[17] F. Yang, M. S. Habibullah, T. Zhang, Z. Xu, P. Lim, and S. Nadarajan, "Health index-based prognostics for remaining useful life predictions in electrical machines," *IEEE Trans. Ind. Electron.*, vol. 63, no. 4, pp. 2633–2644, Apr. 2016.

[18] D. Wu, C. Jennings, J. Terpenny, R. X. Gao, and S. Kumara, "A comparative study on machine learning algorithms for smart manufacturing: Tool wear prediction using random forests," *J. Manuf. Sci. Eng.*, vol. 139, no. 7, 2017, Art. no. 071018.

[19] L. Liao, W. Jin, and R. Pavel, "Enhanced restricted Boltzmann machine with prognosability regularization for prognostics and health assessment," *IEEE Trans. Ind. Electron.*, vol. 63, no. 11, pp. 7076–7083, Nov. 2016.

[20] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mech. Syst. Signal Process.*, vol. 115, pp. 213–237, 2019.

[21] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long short-term memory network for remaining useful life estimation," in *Proc. IEEE Int. Conf. Prognostics Health Manage.*, 2017, pp. 88–95.

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

[23] G. S. Babu, P. Zhao, and X.-L. Li, "Deep convolutional neural network based regression approach for estimation of remaining useful life," in *Proc. Int. Conf. Database Syst. Adv. Appl.*, 2016, pp. 214–228.

[24] J. Zhu, N. Chen, and W. Peng, "Estimation of bearing remaining useful life based on multiscale convolutional neural network," *IEEE Trans. Ind. Electron.*, vol. 66, no. 4, pp. 3208–3216, Apr. 2019.

[25] J. Deutsch and D. He, "Using deep learning-based approach to predict remaining useful life of rotating components," *IEEE Trans. Syst., Man, Cybern.: Syst.*, vol. 48, no. 1, pp. 11–20, Jan. 2018.

[26] C. Zhang, P. Lim, A. Qin, and K. C. Tan, "Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2306–2318, Oct. 2017.

[27] J. A. Bullinaria, "Recurrent neural networks," *Neural Comput., Lecture*, vol. 12, 2013.

[28] R. Zhao, D. Wang, R. Yan, K. Mao, F. Shen, and J. Wang, "Machine health monitoring using local feature-based gated recurrent unit networks," *IEEE Trans. Ind. Electron.*, vol. 65, no. 2, pp. 1539–1548, Feb. 2018.

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

[30] Z. Chen, R. Zhao, Q. Zhu, M. K. Masood, Y. C. Soh, and K. Mao, "Building occupancy estimation with environmental sensors via CD-BLSTM," *IEEE Trans. Ind. Electron.*, vol. 64, no. 12, pp. 9549–9559, Dec. 2017.

[31] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, "Video summarization with long short-term memory," in *Proc. Eur. Conf. Comput. Vision*, 2016, pp. 766–782.

[32] H. Palangi *et al.*, "Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval," *IEEE/ACM Trans. Audio, Speech Lang. Process.*, vol. 24, no. 4, pp. 694–707, Apr. 2016.

[33] Y. Cheng, H. Zhu, J. Wu, and X. Shao, "Machine health monitoring using adaptive kernel spectral clustering and deep long short-term memory recurrent neural networks," *IEEE Trans. Ind. Informat.*, vol. 15, no. 2, pp. 987–997, Feb. 2019.

[34] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2048–2057.

[35] W. Du, Y. Wang, and Y. Qiao, "Recurrent spatial-temporal attention network for action recognition in videos," *IEEE Trans. Image Process.*, vol. 27, no. 3, pp. 1347–1360, Mar. 2018.

[36] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*.

[37] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transp. Res. C, Emerg. Technol.*, vol. 90, pp. 166–180, 2018.

[38] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[39] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," 2012, *arXiv:1207.0580*.

[40] A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage propagation modeling for aircraft engine run-to-failure simulation," in *Proc. Int. Conf. Prognostics Health Manage.*, 2008, pp. 1–9.

[41] A. Saxena and K. Goebel, "PHM08 challenge data set," NASA AMES Prognostics Data Repository, Moffett Field, CA, USA, Tech. Rep., 2008.

[42] P. Lim, C. K. Goh, K. C. Tan, and P. Dutta, "Estimation of remaining useful life based on switching Kalman filter neural network ensemble," in *Proc. IEEE Int. Conf. Prognostics and Health Management*, 2014, pp. 2–9.

[43] R. Yan, F. Shen, C. Sun, and X. Chen, "Knowledge transfer for rotary machine fault diagnosis," *IEEE Sensors J.*, 2019.

[44] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

Authorized licensed use limited to: Nanyang Technological University. Downloaded on May 12,2021 at 10:28:40 UTC from IEEE Xplore. Restrictions apply.

Zhenghua Chen received the B.Eng. degree in mechatronics engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2011, and the Ph.D. degree in electrical and electronic engineering from Nanyang Technological University, Singapore, in 2017.

He is currently a Scientist with the Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore. His research interests include data analytics in smart buildings, ubiquitous computing, Internet of Things, machine learning, and deep learning.

Min Wu received the B.S. degree in computer science from the University of Science and Technology of China, Hefei, China, in 2006, and the Ph.D. degree in computer science from Nanyang Technological University, Singapore, in 2011.

He is currently a Senior Scientist with Data Analytics Department, Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore. His current research interests include machine learning, data mining, and bioinformatics.

Dr. Wu was the recipient of the Best Paper Awards in the International Conference on Bioinformatics in 2016 and International Conference on. Database Systems for Advanced Applications in 2015. He also won the International Joint Conferences on Artificial Intelligence Competition on repeated buyers prediction in 2015.

Rui Zhao received the B.Eng. degree in measurement and control from Southeast University, Nanjing, China, in 2012, and the Ph.D. degree in machine learning from Nanyang Technological University, Singapore, in 2017.

He is currently working as a Quantitative Researcher with Harveston, Singapore. His current research interests include machine learning and its applications on text mining and machine health monitoring.

Feri Guretno received the B.Eng. degree in electrical and electronic engineering from Nanyang Technological University, Singapore, in 2007, and the M.Sc. degree in system design and management from the National University of Singapore, Singapore, in 2010.

He was the Founding Engineer for iTwin, an A*STAR Startup, that enables simple and secure access to data and networks. He has worked at an early stage of other technological startups in Indonesia and Singapore. He is currently working as a Senior Research Engineer with Machine Intelection Department, Institute for Infocomm Research, Singapore, focusing in Industrial Internet of Things and medical technology.

Xiaoli Li received the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences. He is currently a Principal Scientist with the Institute for Infocomm Research, A*STAR, Singapore. He is also an Adjunct Professor with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. He has authored and coauthored more than 180 high quality papers. His research interests include data mining, machine learning, artificial intelligence, and bioinformatics.

Prof. Li has been serving as a (Senior) Program Committee Member/Workshop Chair/Session Chair in leading data mining and artificial-intelligence-related conferences, including KDD, ICDM, SDM, PKDD/ECML, WWW, IJCAI, AAAI, ACL, and CIKM. He has won numerous best paper/benchmark competition awards.

Ruqiang Yan (M'07–SM'11) received the M.S. degree in precision instrument and machinery from the University of Science and Technology of China, Hefei, China, in 2002, and the Ph.D. degree in mechanical engineering from the University of Massachusetts, Amherst, MA, USA, in 2007.

From 2009 to 2018, he was a Professor with the School of Instrument Science and Engineering, Southeast University, Nanjing, China. He joined the School of Mechanical Engineering, Xi‘an Jiaotong University, Xi'an, China, in 2018. He holds 28 patents, and authored and coauthored two books and more than 200 papers in technical journals and conference proceedings. His research interests include data analytics, machine learning, and energy-efficient sensing and sensor networks for the condition monitoring and health diagnosis of large-scale, complex, dynamical systems.

Dr. Yan became a Fellow of the American Society of Mechanical Engineers, in 2019. He was the recipient of several honors and awards including the IEEE Instrumentation and Measurement Society Technical Award in 2019, the New Century Excellent Talents in University Award from the Ministry of Education in China, in 2009, and multiple best paper awards. He is an Associate Editor-in-Chief for the IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT and an Associate Editor for the IEEE SYSTEMS JOURNAL.

Authorized licensed use limited to: Nanyang Technological University. Downloaded on May 12,2021 at 10:28:40 UTC from IEEE Xplore. Restrictions apply.