Reliability Engineering and System Safety 172 (2018) 1–11

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

Remaining useful life estimation in prognostics using deep convolution neural networks

Xiang Li [a,*], Qian Ding [b], Jian-Qiao Sun [c]

[a] College of Sciences, Northeastern University, Shenyang 110819, China
[b] Department of Mechanics, Tianjin University, Tianjin 300072, China
[c] School of Engineering, University of California, Merced, CA 95343, USA

ARTICLE INFO

Keywords:
Prognostics and health management
Remaining useful life
Deep learning
Convolution neural network

ABSTRACT

Traditionally, system prognostics and health management (PHM) depends on sufficient prior knowledge of critical components degradation process in order to predict the remaining useful life (RUL). However, the accurate physical or expert models are not available in most cases. This paper proposes a new data-driven approach for prognostics using deep convolution neural networks (DCNN). Time window approach is employed for sample preparation in order for better feature extraction by DCNN. Raw collected data with normalization are directly used as inputs to the proposed network, and no prior expertise on prognostics and signal processing is required, that facilitates the application of the proposed method. In order to show the effectiveness of the proposed approach, experiments on the popular C-MAPSS dataset for aero-engine unit prognostics are carried out. High prognostic accuracy on the RUL estimation is achieved. The superiority of the proposed method is demonstrated by comparisons with other popular approaches and the state-of-the-art results on the same dataset. The results of this study suggest that the proposed data-driven prognostic method offers a new and promising approach.

© 2017 Elsevier Ltd. All rights reserved.

1. Introduction

Engineering maintenance and prognostics are very crucial in many industry areas such as aerospace, manufacturing, automotive, heavy industry and so forth. While traditional strategies such as breakdown corrective maintenance and scheduled preventive maintenance [1] are becoming less capable of meeting the increasing industrial demand of efficiency and reliability, intelligent prognostic and health management (PHM) technologies which is also known as condition-based maintenance (CBM), are showing promising abilities for application in industries [2]. The goals of PHM include maximizing the operational availability, reduction of maintenance costs and improvement of system reliability and safety by monitoring the facility conditions. Remaining useful life (RUL) can be estimated based on history trajectory data, that is very important for improving maintenance schedules to avoid engineering failures and save the resultant costs [3]. This paper proposes a novel deep learning method for RUL estimation.

Generally, the existing methods for PHM can be grouped into three main categories, i.e. model-based approaches [4], data-driven approaches [5] and hybrid approaches [6]. While model-based approaches tend to be more accurate if the complex system degradation is modeled precisely [7], they require extensive prior knowledge about physical systems which is usually unavailable in practice, e.g. aircraft engines. Popular model-based approaches include particle filter [8], Eyring model [9] , Weibull distribution [10] *etc*. On the other hand, the data-driven approaches are able to model the degradation characteristics based on historical sensor data. The underlying correlations and causalities in the collected sensor data can be revealed, and the corresponding system information such as RUL can be inferred. Data-driven approaches usually require sufficient historical data for training models. Since they do not rely on much prior expertise on prognostics and are easy to be generalized, many data-driven algorithms have been proposed in the recent years and good prognostic results have been achieved, including artificial neural network (NN) [11], support vector machine (SVM) [12], hidden Markov models [13] *etc*. By combining model-based and data-driven approaches, the hybrid approaches aim to utilize the advantages of both the approaches and avoid the disadvantages [14]. However, it still remains very challenging to develop an effective hybrid approach, especially for the combination of the physical models and the recently developed data-driven technologies such as deep learning. Therefore, a novel data-driven approach is proposed in this study

In the past years, discovering the relationship between the monitored system data and the corresponding RUL has been receiving increasing attention in data-driven prognostics. A number of machine learning tech-

* Corresponding author.
E-mail address: xiangli@mail.neu.edu.cn (X. Li).

https://doi.org/10.1016/j.ress.2017.11.021
Received 29 June 2017; Received in revised form 16 October 2017; Accepted 26 November 2017
Available online 2 December 2017
0951-8320/© 2017 Elsevier Ltd. All rights reserved.

niques, especially neural network-based approaches, have been developed to learn the mapping from the collected feature data to the associated RUL. The advantage of applying neural networks on prognostic and health management lies in that highly nonlinear, complex, multi-dimensional system can be well modeled without prior expertise on the system physical behavior. Different kinds of system data, such as raw sensor readings, can be directly used as model inputs. While the confidence limits for the RUL predictions can not be naturally provided [15], the neural network-based approaches are promising on prognostic problems.

Huang et al. [16] utilized the traditional multi-layer perceptron (MLP) approach for modeling the remaining useful life of the laboratory-tested bearings, and reported the prediction results were superior to the reliability-based approaches. Tian [17] developed an artificial neural network (ANN) method for estimating the RUL of equipment. The ANN model takes the age and multiple condition monitoring measurement values at the present and previous inspection points as the inputs, and the equipment life percentage as the output. Fink and colleagues [18] proposed a multi-layer neural network with multi-valued neurons approach to deal with the reliability and degradation time series prediction problem, and carried out a case study on predicting the degradation of a railway turnout system. To overcome the drawback that the confidence limits of RUL estimation can not be directly obtained with the NN-based methods in general, Khawaja and colleagues [19] introduced a confidence prediction neural network approach with a confidence distribution node. In addition, fuzzy logic has been integrated into MLP networks to capture more information for PHM by many researchers [20]. Malhi et al. [21] proposed a competitive learning-based approach to long-term prognostics of machine health status using recurrent neural networks (RNN). Vibration signals from a defect-seeded rolling bearing are pre-processed with the continuous wavelet transform and used as the model inputs. As an improvement of the traditional RNN, a long short term memory (LSTM) based neural network scheme was proposed by Yuan et al. [22] for RUL estimation of aero-engines in the cases of complicated operations, hybrid faults and strong noises. LSTM was also utilized by Zhao et al. [23] for a tool wear health monitoring task.

Recently, deep learning network is emerging as a highly effective network structure for pattern recognition, that holds the potential to improve performance in the current intelligent prognostics. Deep learning is characterized by the deep network architecture where multiple layers are stacked in the network to fully capture the representative information from raw input data [24]. High-level abstractions of data can be modeled well with the help of the complex deep structures, leading to more efficient feature extraction compared with the shallow networks. Deep learning methods have gained great interests and achieved significant results in many fields, including image recognition [25], speech recognition [26] *etc*. Since the raw data obtained from machinery health monitoring share similar high dimensionality with those in image processing researches, deep learning architecture has great potential in PHM and RUL estimation.

Ren et al. [27] proposed an integrated deep learning approach for multi-bearing remaining useful life collaborative prediction by combining both time domain and frequency domain features. Numerical experiments on a real dataset show the effectiveness and superiority of the proposed approach. A new restricted Boltzmann machine (RBM) for representation learning was proposed by Liao et al. [28] to predict RUL of machines, where a new regularization term and unsupervised self-organizing map algorithm are used. Zhang et al. [29] proposed a multi-objective deep belief networks (DBN) ensemble method, where an evolutionary algorithm is integrated with the traditional DBN training technique to evolve multiple DBNs simultaneously subject to accuracy and diversity. State-of-the-art prognostic performance was achieved on the popular benchmarking problem, i.e. NASA's turbofan engine degradation problem [30].

Within the deep learning architecture, convolution neural networks (CNNs), that are specifically designed for variable and complex signals, are further utilized in this study. CNNs have shown remarkable success in various applications in the past few years. CNN was first proposed by LeCun et al. for image processing. Its ability to maintain data information regardless of scale, shift and distortion invariance is presented. A large number of researches on computer vision, speech processing *etc*. have benefited from CNN's characteristics of local receptive fields, shared weights and spatial sub-sampling. Babu et al. [31] built a 2-dimensional (2D) deep convolution neural network to predict the RUL of system based on normalized variate time series from sensor signals, where one dimension of the 2D input is the number of sensors. Average pooling is adopted in their work and a linear regression layer is placed on the top layer. While deep CNN structure has shown great ability on feature extraction, very limited research can be found on its applications on machinery remaining useful life prediction problems. In this study, the CNN structure is employed to extract the local data features through the deep learning network for better prognostics.

A new deep learning architecture for RUL estimation in prognostics is proposed in this paper. Time window approach is employed for sample preparation in order for better feature extraction by CNN. Raw sensor measurements with normalization are directly used as model inputs to the proposed network, and no prior expertise on prognostics and signal processing is required, that facilitates the industrial application of the proposed method. High-level abstract features can be successfully extracted by the deep CNN architecture, and the associated RUL value can be estimated based on the learned representations. Using time window, data normalization and deep CNN structure, the proposed method is expected to obtain higher prognostic accuracy compared with the traditional machine learning methods. Comprehensive analysis of the proposed approach and comparisons with existing methods are presented in this study.

In the recent years, development of modern aeronautical technology leads to a complex aircraft system, where high reliability, quality and safety are required in very harsh environment. The engine is the key component of the aircraft and there is always a pressing need to develop new approaches to better evaluate the engine performance degradation and estimate the remaining useful life [32]. In this paper, the RUL for aero-engines is estimated as a case study and the popular publicly available NASA C-MAPSS dataset [30] is used to validate the effectiveness of the proposed method. Comparisons with the state-of-the-art results on the same dataset show the superiority of the proposed network.

This paper starts with the description of the proposed deep learning structure in Section 2, along with brief introductions of CNN. The proposed method is experimentally validated using the C-MAPSS dataset in Section 3. The effectiveness and superiority of the method are demonstrated by comparisons with other popular methods. We close the paper with conclusions in Section 4.

2. Proposed deep learning architecture

In this section, the proposed deep learning architecture for prognostics is presented, as well as its key components, i.e. convolution neural networks, and dropout technique.

2.1. Convolutional neural network

Convolutional neural networks (CNNs) were first proposed by LeCun for image processing, which has two characteristics, i.e. spatially shared weights and spatial pooling. CNNs have achieved significant success in many research and industry fields including computer vision [25], natural language processing, speech recognition [33] and so forth. The convolutional layers convolve multiple filters with raw input data and generate features, and the following pooling layers extract the most significant local features afterwards. The input data are usually 2-dimensional (2D) data for CNNs to learn abstract spatial features by alternating and stacking convolutional kernels and pooling operation.

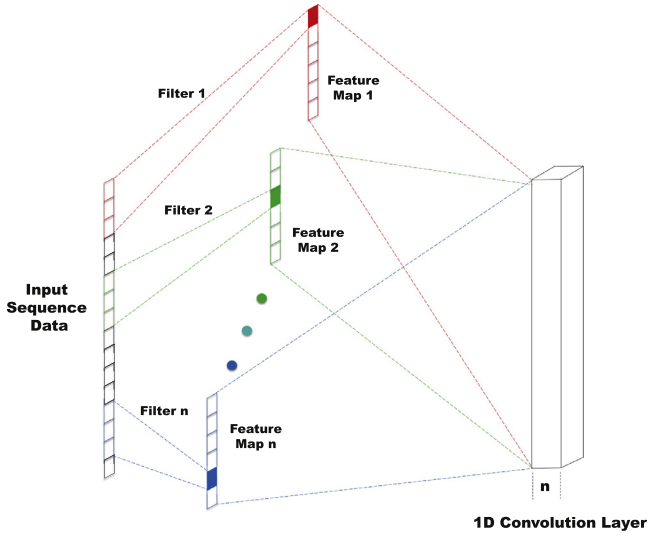

Fig. 1. Illustration for 1D CNN operation.

In this study, the input data is prepared in 2D format where one dimension is the feature number and the other is the time sequence of each feature (The details of data preparation will be presented in Section 3.1). However, considering the collected machinery features are from different sensors in this prognostic problem, the relationship between the spatially neighboring features in the data sample is not remarkable. Therefore, while the input and the corresponding feature maps have 2 dimensions, the convolution filters in the proposed network are 1-dimensional (1D) in practice. In the following, the 1D CNN is briefly introduced and [31,34,35] can be referred to for 2D CNN.

The input 1-dimensional sequential data is assumed to be $\mathbf{x} = [x_1, x_2, \dots, x_N]$ where N denotes the length of the sequence. The convolution operation in the convolutional layer can be defined as a multiply operation between a filter kernel $\mathbf{w}$, $\mathbf{w} \in R^{F_L}$, and a concatenation vector representation $\mathbf{x}_{i:i+F_L-1}$, which can be expressed as,

$$\mathbf{x}_{i:i+F_L-1} = x_i \oplus x_{i+1} \oplus \cdots \oplus x_{i+F_L-1}, \tag{1}$$

where $x_{i:i+F_L-1}$ represents a window of F_L length sequential signal starting from the i-th point, and $\oplus$ concatenates each data samples into a longer embedding. The final convolution operation is defined as,

$$z_i = \varphi(\mathbf{w}^T \mathbf{x}_{i:i+F_L-1} + b), \tag{2}$$

where ^{*T} denotes the transpose of a matrix *, and b and φ represent the bias term and non-linear activation function, respectively. The output z_i can be considered as the learned feature of the filter kernel $\mathbf{w}$ on the corresponding subsequence $\mathbf{x}_{i:i+F_L-1}$. By sliding the filter window from the first point to the last point in the sample data, the feature map of the j-th filter can be obtained, which is denoted as,

$$\mathbf{z}_j = [z_j^1, z_j^2, \dots, z_j^{N-F_L+1}], \tag{3}$$

where j is the j-th filter kernel. In CNNs, multiple filter kernels can be applied in the convolution layer with different filter length F_L. The effect of filter number F_N and length on the network performance will be presented in Section 3.2. The framework for 1D CNN operation is displayed in Fig. 1.

In practical applications, the determinations of the two parameters F_N and F_L depend on the specific task. Based on the current understanding in the literature [25,31,35], larger filter size and number generally lead to higher prediction accuracy. However, the computational burden also becomes heavier. A tradeoff has to be made in real case studies. Therefore, regarding the fault diagnosis study, medium values of the two parameters are preferred, and $F_N = 10$ and $F_L = 10$ are used as the default values.

Usually, a pooling layer is applied to the feature maps generated by the convolutional layer [31]. On the one hand, the pooling is able to extract the most significant local information in each feature map. On the other hand, the feature dimensionality, i.e. the number of model parameters, can be remarkably reduced by this operation. Therefore, pooling is well suited for very high-dimensional problems such as image processing. However, while computing efficiency can be improved by this operation, noticeable useful information is filtered to some extent. Hence, despite the popular employment of pooling in convolution neural network, pooling is not suggested in this prognostic problem where the raw feature dimension is comparatively low.

2.2. Dropout

Dropout is a technique that can help to reduce data overfitting when training a neural network especially with a small training dataset [36]. Training data overfitting generally results in excellent network performance on the training dataset and poor performance on the testing dataset. Dropout provides an easy and effective way to solve this problem. In this study, the dropout technique is applied on the proposed network to prevent complex co-adaptations on the training data and avoid the extraction of the same features repeatedly.

In practice, dropout can be realized by setting the activated outputs of some hidden neurons to zero so that the neurons are not included in the forward propagation training process. However, dropout is turned off in the testing process, that indicates all the hidden neurons are involved during testing. In this way, the robustness of the network is enhanced. Dropout can be also considered as a simple approach for model ensemble within the network, that helps to improve the feature extraction capability of the network.

2.3. Proposed network structure

Deep neural networks are able to adaptively capture the representation information from raw input signals through multiple non-linear transformations and approximate complex non-linear functions, and are used as the main architecture in this study. In general, the proposed deep learning method consists of two sub-structures, i.e. multiple convolution neural networks and fully-connected layer for regression. Fig. 2 shows the architecture of the proposed method for RUL estimation.

First, the input data sample is prepared in 2-dimensional (2D) format, that facilitates the application of convolution operation. The dimension of the input is $N_{tw} \times N_{ft}$, where N_{tw} denotes the time sequence dimension and N_{ft} is the number of selected features. The raw features are usually obtained from multiple sensor measurements. The details of the data preparation will be presented in Section 3.1.

Next, 4 convolution layers are stacked in the network for feature extraction. The 4 layers have the same configuration that F_N filters are used and the filter size is $F_L \times 1$. Zeros-padding operation is implemented to keep the feature map dimension unchanged [37]. So far, the obtained output is F_N feature maps whose dimension is $N_{tw} \times N_{ft}$ that is the same with the original input sample. We use another convolution layer with 1 filter to combine the previous feature maps to be a unique one. The filter size is 3×1. In this way, the high-level representation for each raw feature is obtained.

Afterwards, the 2-dimensional feature map is flattened and connected with a fully-connected layer. Note that dropout technique is used on the last feature map, i.e. the flattened layer, to relieve overfitting. Finally, one neuron is attached at the end of the proposed network for RUL estimation.

All the layers use *tanh* as the activation functions, and *Xavier* normal initializer is employed for the weight initializations [38]. To further improve the prognostic performance, a fine-tuning process using the back-propagation (BP) algorithm is applied [39], where the parameters of the proposed model are updated to minimize the training error. The *Adam* algorithm [40] is employed for optimization.

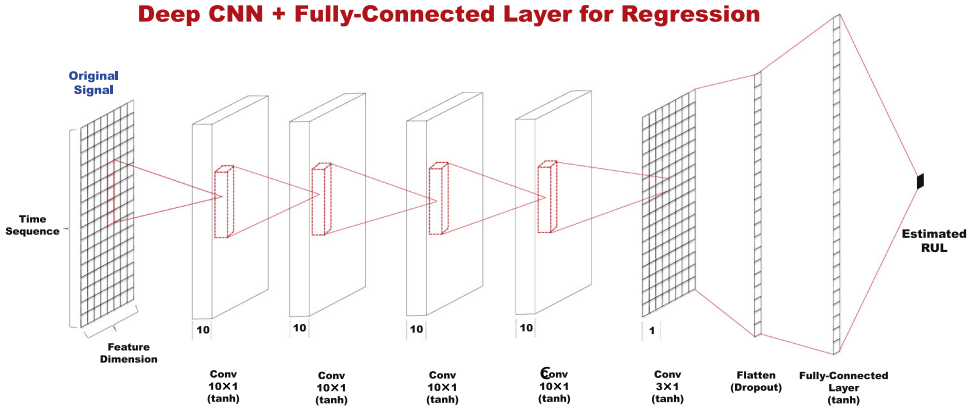

Fig. 2. Proposed deep learning architecture for RUL estimation.

It should be pointed out that, while 2D convolution neural network is used for feature extraction, the convolution operation is actually carried out in 1 dimension, i.e. the time sequence dimension for each feature. Therefore, at first, the multiple stacked convolution layers aim to learn the high-level representations for each raw feature respectively, and the fully-connected layer uses all the learned representations of the features for the final regression. Comparing with most existing deep CNN methods on prognostics which try to learn the spatial relationships of different features at the beginning and further extract information from the learned abstract representation with many layers, the proposed method is more suitable for the extraction of features from different sensor measurements.

3. Experimental study

3.1. Experimental setup

3.1.1. C-MAPSS Dataset

In this paper, the proposed method is evaluated on a prognostic benchmarking problem, i.e. NASA's turbofan engine degradation problem [30,41]. This popular dataset contains simulated data produced by a model-based simulation program, i.e. Commercial Modular Aero-Propulsion System Simulation (C-MAPSS), which was developed by NASA. The C-MAPSS dataset includes 4 sub-datasets that are composed of multi-variate temporal data obtained from 21 sensors. Each sub-dataset contains one training set and one test set. The training datasets include run-to-failure sensor records of multiple aero-engines collected under different operational conditions and fault modes. Each engine unit starts with different degrees of initial wear and manufacturing variation that is unknown and considered to be healthy. As time progresses, the engine units begin to degrade until they reach the system failures, i.e. the last data entry corresponds to the time cycle that the engine unit is declared unhealthy. On the other hand, the sensor records in the testing datasets terminate at some time before system failure, and the goal of this task is to estimate the remaining useful life of each engine in the test dataset. For verification, the actual RUL value for the testing engine units are also provided. In this study, a comprehensive evaluation of the proposed method is carried out on all the four sub-datasets. The detailed information of the C-MAPSS dataset is presented in Table 1, and the four sub-datasets are denoted as FD001, FD002, FD003 and FD004 respectively.

In the training process, all the available engine measurement data points are used as the training samples, and each data point is associated with its RUL label as the target. A piecewise linear degradation model [5] is used to obtain the RUL label with respect to each training sample. During testing, the one data point corresponding with the last recorded cycle for each engine unit is generally used as the testing sample. The actual RUL of the testing samples are provided in the dataset.

Table 1
Information of the C-MAPSS dataset.

	C-MAPSS			
Dataset	FD001	FD002	FD003	FD004
Engine units for training	100	260	100	249
Engine units for testing	100	259	100	248
Operating conditions	1	6	1	6
Fault modes	1	1	2	2
Training samples (default)	17,731	48,819	21,820	57522
Testing samples	100	259	100	248

3.1.2. Data pre-processing

The multi-variate temporal data in the C-MAPSS dataset contains engine unit measurements from 21 sensors [30]. However, some sensor readings have constant outputs in the engine's lifetime and they do not provide valuable information for RUL estimation. Therefore, 14 sensor measurements out of the total 21 sensors are used as the raw input features as did in the literature [5,29], whose indices are 2, 3, 4, 7, 8, 9, 11, 12, 13, 14, 15, 17, 20 and 21.

For each of the 4 sub-datasets in C-MAPSS, the collected measurement data from each sensor are normalized to be within the range of $[-1, 1]$ using the min-max normalization method,

$$x_{norm}^{i,j} = \frac{2(x^{i,j} - x_{\min}^{j})}{x_{\max}^{j} - x_{\min}^{j}} - 1, \quad \forall i, j, \tag{4}$$

where $x^{i,j}$ denotes the original i-th data point of the j-th sensor, and $x_{norm}^{i,j}$ is the normalized value of $x^{i,j}$. $x_{\max}^{j}$ and $x_{\min}^{j}$ denote the maximum and minimum values of the original measurement data from the j-th sensor, respectively.

Different from common regression problems, the desired output value of the input data is difficult to determine for a remaining useful life prediction problem. That is because in many industrial applications, it is impossible to evaluate the precise health condition and estimate the RUL of the system at each time step without an accurate physics-based model [29]. For this popular dataset, a piece-wise linear degradation model has been validated to be suitable and effective [42]. In general, the engine unit works normally in the early age and degrades linearly afterwards. It is assumed to have a constant RUL label in the initial period. Following the recent researches in the literature [2,5,42], R_{early} which is a constant RUL value, is used as the target labels for the data points in the early period. It should be noted that R_{early} has noticeable effect on the prognostic performance on the dataset. By comparing with existing related works, we demonstrate the effectiveness of the proposed method with suitable R_{early} in the following Section 3.2.

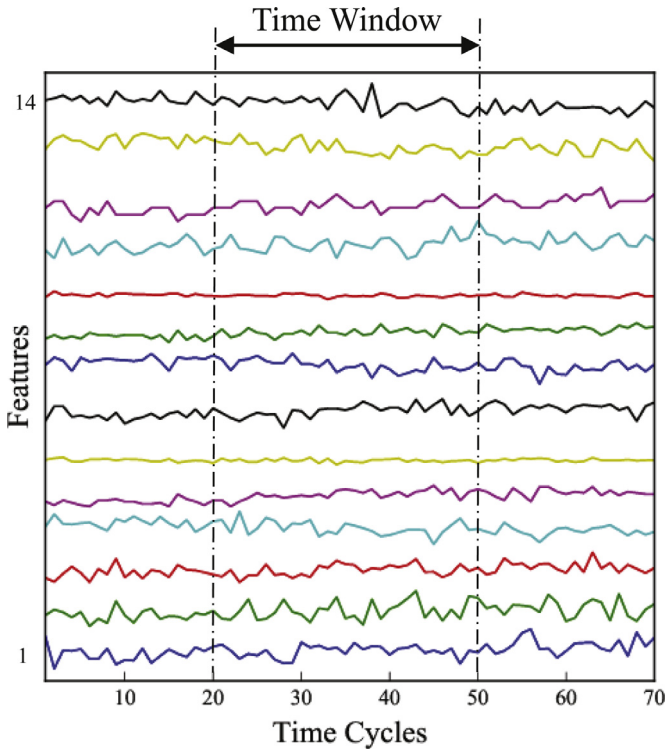

Fig. 3. Illustration of one training sample with 14 selected features within a time window of length 30.

3.1.3. Time window processing

In the multi-variate time series-based problems such as RUL estimation, more information can be generally obtained from the temporal sequence data compared with the multi-variate data point sampled at a single time step. Time sequence processing has a large potential for better prediction performance. In this paper, a time window is adopted for the data preparation to use the multi-variate temporal information.

Let N_{tw} denote the size of the time window. At each time step, all the past sensor data within the time window are collected to form a high-dimensional feature vector, and used as the inputs for the network.

Fig. 3 shows one normalized data sample from the 14 selected sensors within a time window of size 30 with respect to a single engine unit in the training sub-dataset FD001. The shape of the data sample corresponds with the input size of the proposed network, as presented in Fig. 2. The effect of time window on the network performance will be discussed in Section 3.2.3.

3.1.4. Performance metrics

In this study, 2 metrics have been used for evaluating the performance of the proposed prognostic method, i.e. scoring function and root mean square error.

The scoring function used in this study has been proposed by many researchers [2,43,44] and also employed by the International Conference on Prognostics and Health Management Data Challenge. The function is illustrated in Eq. (5).

$$s = \sum_{i=1}^{N} s_i,$$
$$s_i = \begin{cases} e^{-\frac{d_i}{13}} - 1, & \text{for } d_i < 0, \\ e^{\frac{d_i}{10}} - 1, & \text{for } d_i \geq 0, \end{cases} \tag{5}$$

where s denotes the score and N is the total number of testing data samples. $d_i = RUL_i' - RUL_i$, that is the error between the estimated RUL value and the actual RUL value for the i-th testing data sample. The scoring function penalizes late prediction more than early prediction, that is because late prediction usually leads to more severe consequences in many fields such as aerospace industries.

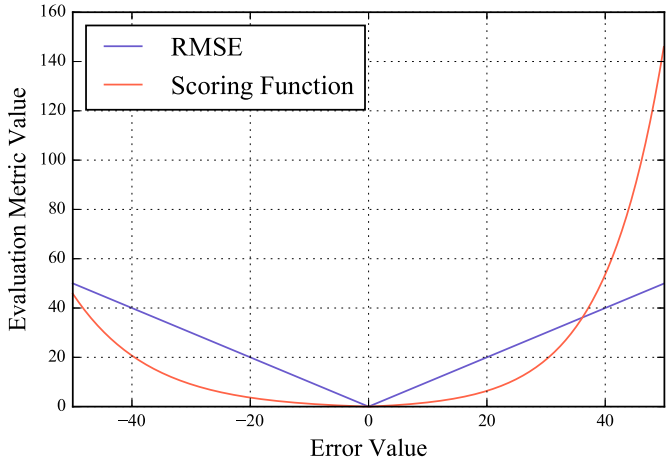

Fig. 4. Comparison between the scoring function and RMSE with respect to different error values.

Another popular metric to evaluate the effectiveness of the proposed method is Root Mean Square Error (RMSE). The formulation of RMSE is as follows,

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} d_i^2}. \tag{6}$$

Fig. 4 shows the comparison between the two evaluation metrics.

3.1.5. Prognostic procedure

The flow chart of the proposed prognostic method is presented in Fig. 5. First, the C-MAPSS sub-datasets are pre-processed that 14 raw sensor measurements are selected and the corresponding data are normalized to be within the range $[-1, 1]$. Then the datasets for training and testing are prepared with each sample containing the time sequence information within the time window of N_{tw} length. It should be noted that the normalized data, which are prepared in 2D format, are used directly as the model input. No hand-crafted signal processing feature is needed, such as skewness, kurtosis *etc*. Therefore, no prior expertise on prognostics and signal processing is required in the proposed method.

Next, based on the specific signal processing problem and the dataset information, the proposed deep convolution neural network (DCNN) for RUL estimation starts to be built, and its configuration is determined including the number of hidden layers, convolution filter number and length *etc*. The DCNN takes as the inputs the normalized training data, and the labeled RUL values for the training samples are used as the target outputs of the network.

Back-propagation learning is used for the updates of the weights in the network. The Adam optimization algorithm is used with mini-batches for the updates. For each training epoch, the samples are randomly divided into multiple mini-batches with each batch containing 512 samples, and put into the training system. Next, the network information, i.e. the weights in each layer, are optimized based on the mean loss function of each mini-batch. It should be noted that the selection of batch size affects the network training performance [45]. The batch size of 512 samples is found appropriate based on the experiments and it is used in all the case studies in this paper. In addition, varying learning rate is adopted. For the first 200 epochs from the beginning, the learning rate is 0.001 for fast optimization. The learning rate of 0.0001 is used afterwards for stable convergence. The maximum number of the training epochs is 250 by default.

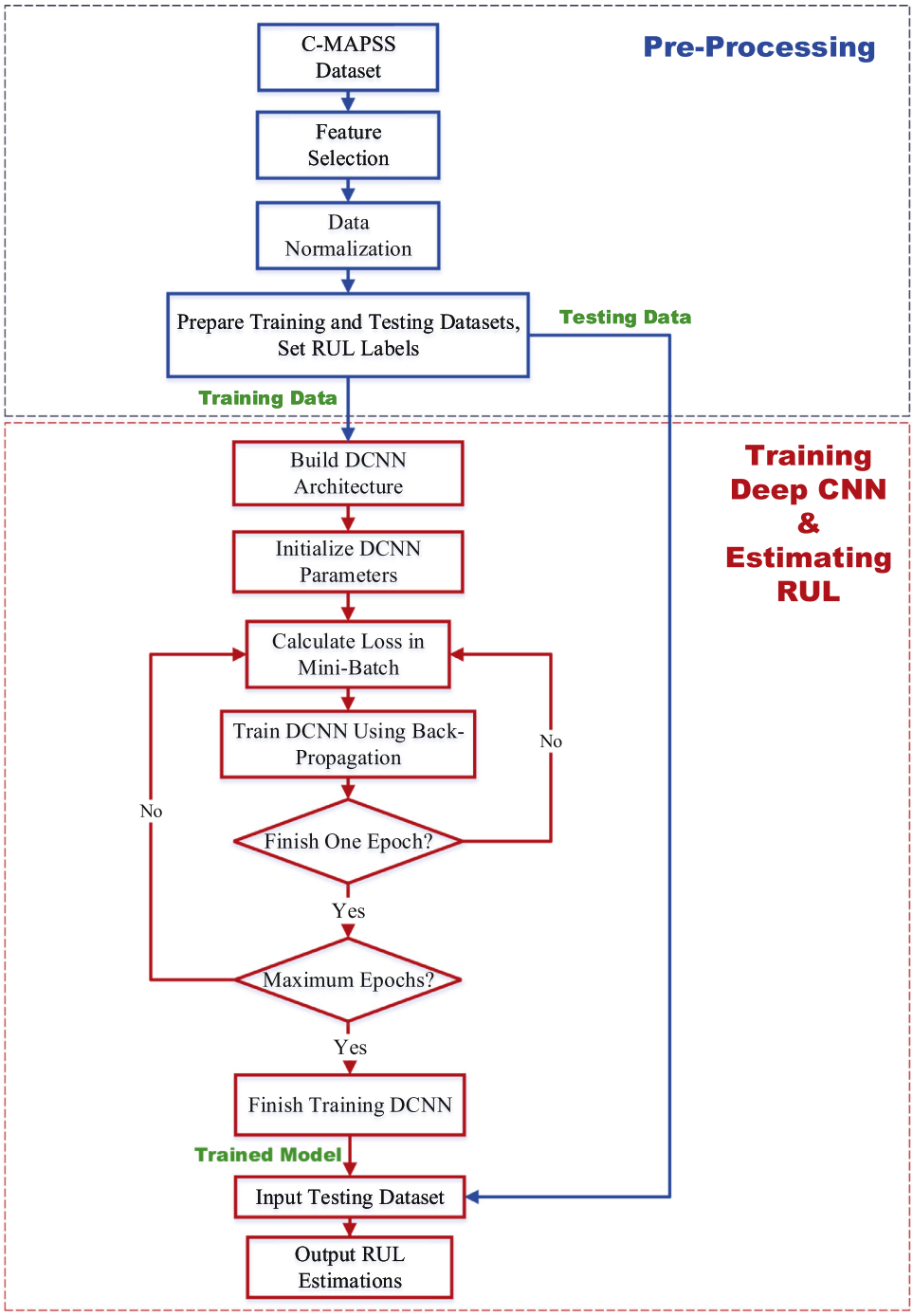

Fig. 5. Flow chart of the proposed method for prognostics.

Table 2
Default parameters of the proposed method and the experimental setting.

Parameter	Value	Parameter	Value
F_N	10	N_{tw} for FD001 to FD004	30/20/30/15
F_L	10	Neurons in fully-connected layer	100
N_{ft}	14	Dropout rate	0.5
R_{early}	125	Convolution Layers	5
Batch size	512	Epoch number	250

Finally, the testing data samples are fed into the trained network for the RUL estimations, and the prognostic accuracy can be obtained. The default parameters of the proposed method are presented in Table 2.

3.1.6. Compared approaches

The proposed deep learning framework is able to provide a systematic and accurate method of prognostics. In order to show the superiority of the proposed method to existing popular methods, its testing performance is compared with those of other network architectures in this paper. Different prognostic methods for RUL estimation are carried out, including basic neural network (NN), deep neural network (DNN), recurrent neural network (RNN) and long short-term memory (LSTM).

1. NN
   The basic neural network, which is also known as multi-layer perceptron (MLP), is used for comparison with 1 hidden layer of 500 neurons, that is a reasonable number in neural network-based ap-

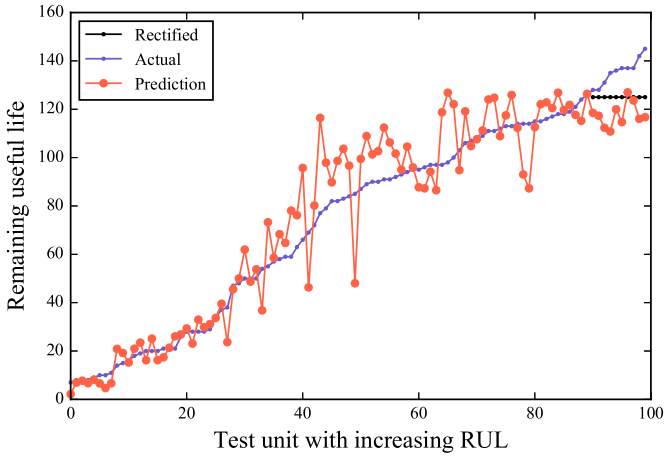

Fig. 6. Sorted prediction for the 100 testing engine units in FD001.

proaches. Dropout technique which can significantly enhance the generalization ability of neural network is adopted, and the dropout rate of 0.5 is used.

2. DNN
   The deep neural network has 4 hidden layers. The number of neurons in the hidden layers is 500, 400, 300 and 100, respectively. DNN shares similar depth with the proposed network and provides a relatively fair comparison, considering deeper architecture is usually difficult to be effectively trained. Dropout is also employed in each hidden layer.
3. RNN
   The recurrent neural network contains feedback connections from the hidden or output layers to the preceding layers [21], thus having the ability of processing dynamic information. RNN is a more effective model that involves time-series data. Instead of using the convolution layers in the proposed DCNN for further feature extraction, we adopt 5 recurrent layers in the RNN network for comparison. In this way, the two methods have the same depth, and also similar computational burden according to the experiments that will be presented in the following sections. A fully-connected layer is also attached in order to have similar architecture with the proposed method.
4. LSTM
   Due to the vanishing gradient problem during back-propagation for model training, traditional RNN may not capture long-term dependencies. Therefore, as a variant of RNN, long-short term memory method is preferred by many researchers to prevent back-propagated errors from vanishing or exploding [46]. Gates are introduced in LSTM to enable each recurrent unit to adaptively capture dependencies of different time scales. In order to share similar structure with the proposed network, 5 LSTM layers and 1 fully-connected layer are used for comparison.

For all the comparing methods in this study, the input and output layers are the same with the proposed network, and RMSE is used as the loss function. Back-propagation is employed for the updates of model parameters where the Adam optimization algorithm is used.

3.2. Experimental results and performance analysis

In this section, the prognostic performance of the proposed method for RUL estimation is presented. The effects of different factors on the results are investigated, including the number of hidden layers and time window length. The comparisons with other popular neural network architectures are carried out to show the effectiveness of the proposed structure. Furthermore, the superiority of the proposed approach is demonstrated by comparing with the latest state-of-the-art prognostic results on the same C-MAPSS dataset.

In this paper, the reported experimental results are averaged by 10 trials to reduce the effect of randomness, and the mean values and standard deviations are provided. All the experiments are carried out on a PC with Intel Core i7 CPU, 8-GB RAM and GEFORCE GTX 950M GPU.

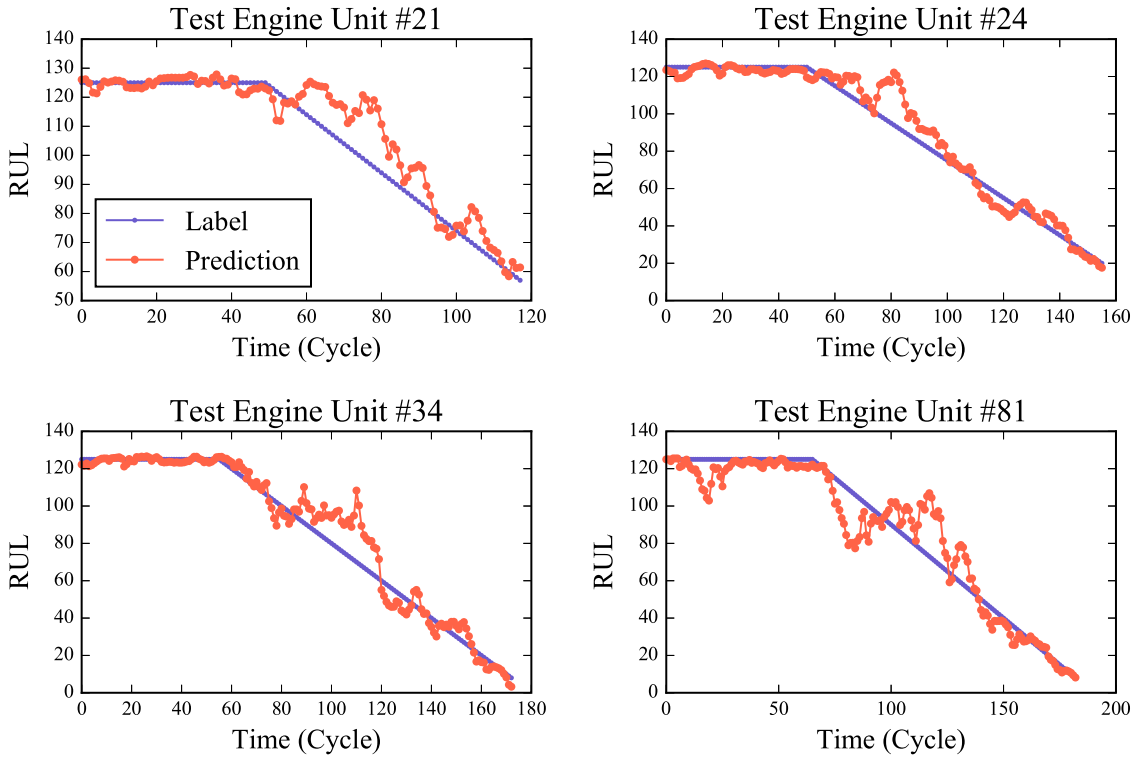

Fig. 7. Four examples of life-time RUL predictions for the testing engine units in FD001.

Table 3
Performance comparisons of different methods on the C-MAPSS dataset. STD: Standard deviation.

		NN		DNN		RNN		LSTM		DCNN	
		Mean	STD	Mean	STD	Mean	STD	Mean	STD	Mean	STD
FD001	RMSE	14.80	0.31	13.56	0.21	13.44	0.43	13.52	0.61	**12.61**	0.19
	Score	496.3	14.3	348.3	17.5	339.2	29.0	431.7	42.4	**273.7**	24.1
FD002	RMSE	25.64	0.34	24.61	0.33	24.03	0.26	24.42	0.45	**22.36**	0.32
	Score	18,255	1402	15,622	872	14,245	622	14,459	815	**10412**	544
FD003	RMSE	15.22	0.34	13.93	0.34	13.36	0.38	13.54	0.29	**12.64**	0.14
	Score	522.3	17.1	364.3	19.3	315.7	24.2	347.3	28.0	**284.1**	26.5
FD004	RMSE	25.80	0.44	24.31	0.24	24.02	0.41	24.21	0.36	**23.31**	0.39
	Score	20,422	1231	16,223	895	13,931	1102	14,322	1043	**12466**	853

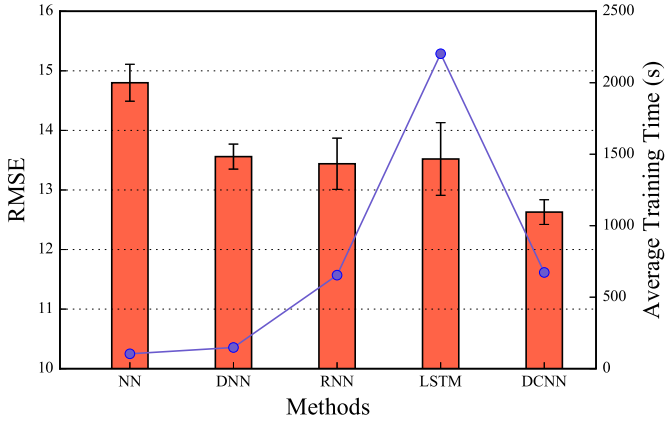

Fig. 8. Prognostic performance by different methods on FD001.

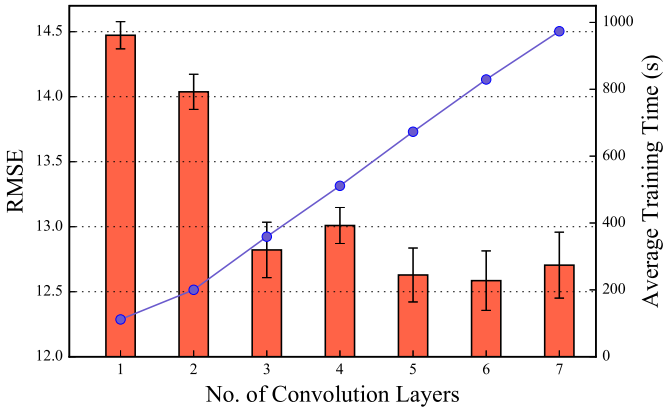

Fig. 9. The effect of the number of convolution layers in the proposed network on the prognostic performance and computing time for the training process on FD001.

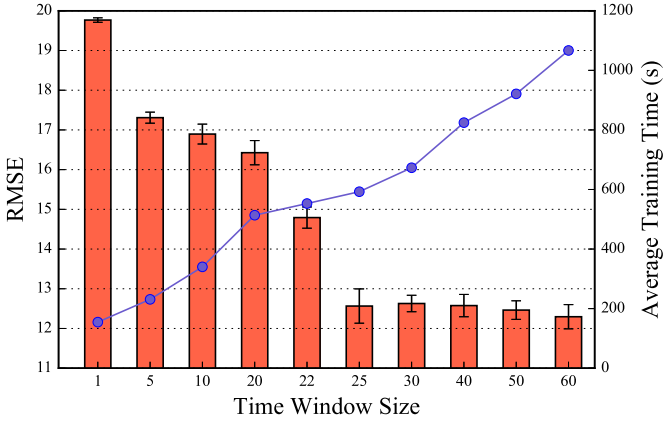

Fig. 10. The effect of the time window size on the prognostic performance and computing time for the training process on FD001.

3.2.1. *Prognostic performance*

The RUL prediction results of the testing engine units in FD001 regarding the last recorded data point are presented in Fig. 6. The testing engine units are sorted by labels from small to large for better observation and analysis. It can be observed that the predicted RUL values by the proposed method are close to the actual values generally. Especially, the prognostic accuracy tends to be higher in the region where the RUL value is small. That is because when the engine unit is working close to failure, the fault feature is enhanced and that can be captured by the proposed network for better prognostics.

Furthermore, the RUL estimations for the life-time of the testing engine units in FD001 before the last recorded cycle are shown in Fig. 7. 4 examples out of one hundred testing engine units, whose unit number are 21, 24, 34 and 81 respectively, are presented for demonstrations. One can notice that the RUL estimations of the last parts of the engine unit life-time are not shown. That is due to the fact that in the testing dataset, the last parts of the sensor measurements are not provided in order to exam the prognostic performance. The actual RUL values for the last recorded cycles are given in the dataset, and the corresponding RUL labels for the previous life-time can be obtained accordingly.

It can be observed that in the early periods in all the 4 cases, the proposed method manages to estimate the RUL values as close to the constant R_{early}. Afterwards, the estimations are almost linearly decreasing with time until the end of the available testing samples. Specifically, despite some noticeable error existing between the predictions and the actual RUL values in general, the prognostic accuracy is high especially when the engine units are close to failure. That is of industrial value since the late period in the engine life-time is very critical for the health management. A good evaluation of the engine status in the late period is able to enhance operation reliability and safety, reduce maintenance costs and improve the whole system performance.

3.2.2. *Comparing with other architectures*

The comprehensive comparison results of the prognostic performance using different methods are presented in Table 3 and Fig. 8. The effectiveness of the proposed method is examined with the default experimental setting on the four sub-datasets. It can be observed that generally, the proposed deep learning method achieves the best performance in all the cases.

The experimental results indicate that the proposed deep convolution neural network architecture is well suited for the prognostic problem. The stacked convolution layers contribute to the learning capability of the network. The RNN structure is the second best using the recurrent information flow. While LSTM is a more advanced variant of RNN, its performance is not as good as RNN in this case study. However, it can be further optimized to get better results [22,47]. The basic neural networks NN and DNN are also competitive. That indicates the

Table 4
Performance comparisons of the proposed method and the latest related papers on the C-MAPSS dataset.

Method	RMSE				
	FD001	FD002	FD003	FD004	R_{early}
Echo state nNetwork with Kalman filter [49]	63.46	N/A	N/A	N/A	Not applied
Support vector machine classifier [50]	29.82	N/A	N/A	N/A	Not applied
Random forest [29]	17.91	29.59	20.27	31.12	Not applied
Gradient boosting [29]	15.67	29.09	16.84	29.01	Not applied
First attempt of deep CNN [31]	18.45	30.29	19.82	29.16	Not provided
Time window based NN [51]	15.16	N/A	N/A	N/A	Not provided
Multi-objective deep belief networks ensemble [29]	15.04	25.05	12.51	28.66	Not applied
RULCLIPPER [42]	13.27	22.89	16.00	24.33	135
LSTM [47]	12.81	N/A	N/A	N/A	125
Proposed method with rectified labels	**12.61**	**22.36**	**12.64**	**23.31**	125
Proposed method without rectified labels	**13.32**	**24.86**	**14.02**	**29.44**	Not applied

sample preparations with raw feature selection, data pre-processing and application of time window are efficient for further feature extraction. Traditionally, the DNN with 4 hidden layers suffer from the overfitting problem. With the help of the regularization technique, i.e. dropout, good prognostic performance is obtained in this study.

Additionally, the RMSEs and scores of FD002 and FD004 are relatively higher due to the increased difficulties of the sub-datasets. On the one hand, the increased number of operating conditions and fault modes make the prognostic problem more complicated. On the other hand, overfitting has large opportunity to occur with the increased number of the engine units.

In summary, the comparison results presented above suggest the proposed method is promising for prognostic problems and able to provide reliable RUL estimations in different cases. Next, the performance analysis of the proposed network is presented.

3.2.3. Effects of number of convolution layers and time window

In this section, the influence of the key parameters in the proposed method is examined, and the sub-dataset FD001 is used for the evaluation for instance.

Fig. 9 shows the effect of the number of convolution layers on the network prognostic performance. It can be observed that generally, more hidden convolution layers lead to lower RMSE values. That indicates the deep architecture is able to capture more useful information than the shallow ones. On the other hand, while higher prognostic accuracy can be obtained by deeper structure, the computing time for the training process increases almost linearly with the hidden layer number. It is noted that the network with 5 convolution layers achieves good performance with medium computing burden, and that is used as the default hidden layer number of the proposed architecture in this paper.

Another important coefficient in the proposed method is the time window size in the sample preparation. We present the effect of the time window size on the network performance in Fig. 10. It should be noted that in the testing set, the recorded data cycles for the testing engine units have different length, and the shortest one has only 31 cycles specifically. In order to provide a more comprehensive analysis of the time window, the testing engine units which have shorter recorded cycles than N_{tw} are removed in the corresponding cases.

It is clearly showed in the figure that larger time window size results in better RUL estimations. More raw information can be covered by larger time window, which is the basis for further feature extraction. It is also observed that there is significant reduction in RUL estimation error when N_{tw} increases from 20 to 30. No remarkable further improvement in the prognostic performance is achieved when N_{tw} exceeds 30. Similar with the effect of the hidden layer number, the computing load for the training process rises as the time window size increases. In addition, as stated previously, the presented results can not represent the whole testing dataset when N_{tw} is larger than 31. However, the general display patterns of the time window effect can be reflected.

Based on the experiments and dataset information, $N_{tw} = 30$ is used as the default setting in the sub-dataset FD001, and the values of N_{tw} in the other sub-datasets can be determined accordingly. Specifically, since the maximum number of the record cycles in FD002, FD003 and FD004 are 21, 38 and 19 respectively, $N_{tw} = 20$, 30 and 15 are used for the sub-datasets correspondingly.

3.2.4. Comparing with related works

The C-MAPSS dataset used in this paper is very popular in prognostic researches, and many state-of-the-art results have been reported in the recent years. Table 4 summaries the latest research results on the four sub-datasets of C-MAPSS. It can be observed that many neural network-based approaches have shown their merits on this prognostic problem, including LSTM, CNN, DBN *etc*.

The proposed deep learning method has achieved promising performance compared with the state-of-the-art results. It should be noted that in this paper, we artificially set an RUL threshold R_{early} for the healthy condition, which has noticeable effect on the experimental performance. The prognostic results without the threshold are also provided. Despite the increase of the RMSE value of the RUL estimation, the results of the proposed method are still competitive. On the other hand, the constant RUL value for the early period is widely used in the corresponding researches [2,5,42]. However, the detailed information of the threshold is reported in limited papers. While different R_{early} is employed in different studies, the presented results in Table 4 are still able to provide general comparisons of the advanced approaches.

In addition, while LSTM algorithm such as in [47] is usually preferred in many tasks for sequential signal processing, its computational burden is relatively high with similar network depth as shown in Fig. 8. The proposed method is able to achieve the prognostic results of the same level with LSTM, and has more simple architecture and lower computing load. Therefore, the proposed method is promising in prognostic tasks.

4. Conclusions

In this paper, a new deep learning method for prognostics is proposed based on convolution neural networks. Dropout technique is employed to relieve overfitting problem. Experiments are carried out on the popular C-MAPSS dataset to show the effectiveness of the proposed method. The goal of the task is to estimate the remaining useful life of aero-engine units accurately. With raw feature selection, data preprocessing and sample preparation using time window, good prognostic performance is achieved with the proposed method, and small error between the prediction and the actual RUL value is obtained for the testing data. The RUL in the life-time of the engine units can be well predicted, especially for the late period close to failure.

Based on the comparisons with other popular neural network structures, it is noted that the proposed deep convolution architecture is well

suited for the prognostic problem. The effects of the number of hidden layers and time window size on the prognostic performance are investigated. Considering the estimation accuracy, computing burden for the training process and the dataset information, the time window with $N_{tw} = 30$ is used for the sample preparation in FD001, and five convolution layers are adopted in the proposed network.

Furthermore, the prognostic results obtained by the proposed method are compared with the state-of-the-art results on the same dataset in the literature. The proposed network has shown its superiority on the prognostic accuracy, and is promising for industrial applications.

Additionally, it should be noted that the data pre-processing method is used in this study, since normalized data facilitate the network training process. However, in real-world online applications, the whole life-span data are not available. One solution for the real time application is that, based on the fact that the maximum and minimum values of each sensor measurement are close to each other for different engine units, the mean values of the maximum and minimum values for each feature in the training dataset can be directly used for the testing units for pre-processing. An alternative way is to adopt the batch normalization technique that has been widely applied in deep learning tasks in the literature [48]. The data normalization is carried out on each hidden layer in the network within the mini-batch, and that is able to accelerate the training process.

While good experimental results have been obtained by the proposed method, further architecture optimization is still necessary, since the current training time is longer than most shallow networks in the literature. Deep learning methods generally suffer from high computing load, and that will be focused on in further research. In addition, the score function is suitable to be applied on the prognostic problem like the C-MAPSS dataset. Efforts should be made on including the score function into the loss function of the neural network in the future.

Acknowledgments

The first author would like to thank Dr. Wen-Qi Ren for the valuable advices on this paper. The material in this paper is based on work supported by grant with number 02060022117047 from Northeastern University, and grants (11172197, 11332008, and 11572215) from the National Science Foundation of China.

References

[1] Azadeh A, Asadzadeh SM, Salehi N, Firoozi M. Condition-based maintenance effectiveness for series-parallel power generation system - a combined Markovian simulation model. Reliabil Eng Syst Saf 2015;142:357–68.
[2] Zhao Z, Bin L, Wang X, Lu W. Remaining useful life prediction of aircraft engine based on degradation pattern learning. Reliabil Eng Syst Saf 2017;164:74–83.
[3] Lee J, Wu F, Zhao W, Ghaffari M, Liao L, Siegel D. Prognostics and health management design for rotary machinery systems - reviews, methodology and applications. Mech Syst Signal Process 2014;42(1–2):314–34.
[4] Pecht M, Gu J. Physics-of-failure-based prognostics for electronic products. Trans Inst Meas Control 2009;31(3–4):309–22.
[5] Heimes FO. Recurrent neural networks for remaining useful life estimation. In: Proceedings of international conference on prognostics and health management; 2008. p. 1–6.
[6] Heng A, Zhang S, Tan ACC, Mathew J. Rotating machinery prognostics: state of the art, challenges and opportunities. Mech Syst Signal Process 2009;23(3):724–39.
[7] Qian Y, Yan R, Gao RX. A multi-time scale approach to remaining useful life prediction in rolling bearing. Mech Syst Signal Process 2017;83:549–67.
[8] Jouin M, Gouriveau R, Hissel D, Pra M-C, Zerhouni N. Particle filter-based prognostics: review, discussion and perspectives. Mech Syst Signal Process 2016;7273:2–31.
[9] Jouin M, Gouriveau R, Hissel D, Pra M-C, Zerhouni N. Degradations analysis and aging modeling for health assessment and prognostics of PEMFC. Reliabil Eng Syst Saf 2016;148:78–95.
[10] Ali JB, Chebel-Morello B, Saidi L, Malinowski S, Fnaiech F. Accurate bearing remaining useful life prediction based on Weibull distribution and artificial neural network. Mech Syst Signal Process 2015;5657:150–72.
[11] Gebraeel N, Lawley M, Liu R, Parmeshwaran V. Residual life predictions from vibration-based degradation signals: a neural network approach. IEEE Trans Ind Electron 2004;51(3):694–700.
[12] Benkedjouh T, Medjaher K, Zerhouni N, Rechak S. Remaining useful life estimation based on nonlinear feature reduction and support vector regression. Eng Appl Artif Intell 2013;26(7):1751–60.
[13] Dong M, He D. A segmental hidden semi-Markov model (HSMM)-based diagnostics and prognostics framework and methodology. Mech Syst Signal Process 2007;21(5):2248–66.
[14] Baraldi P, Compare M, Sauco S, Zio E. Ensemble neural network-based particle filtering for prognostics. Mech Syst Signal Process 2013;41(1–2):288–300.
[15] Sikorska JZ, Hodkiewicz M, Ma L. Prognostic modelling options for remaining useful life estimation by industry. Mech Syst Signal Process 2011;25(5):1803–36.
[16] Huang R, Xi L, Li X, Liu CR, Qiu H, Lee J. Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods. Mech Syst Signal Process 2007;21(1):193–207.
[17] Tian Z. An artificial neural network method for remaining useful life prediction of equipment subject to condition monitoring. J Intell Manuf 2012;23(2):227–37.
[18] Fink O, Zio E, Weidmann U. Predicting component reliability and level of degradation with complex-valued neural networks. Reliabil Eng Syst Saf 2014;121:198–206.
[19] Khawaja T, Vachtsevanos G, Wu B. Reasoning about uncertainty in prognosis: a confidence prediction neural network approach. In: Proceedings of Annual Meeting of the North American Fuzzy Information Processing Society; 2005. p. 7–12.
[20] Wang WQ, Golnaraghi MF, Ismail F. Prognosis of machine health condition using neuro-fuzzy systems. Mech Syst Signal Process 2004;18(4):813–31.
[21] Malhi A, Yan R, Gao RX. Prognosis of defect propagation based on recurrent neural networks. IEEE Trans Instrum Meas 2011;60(3):703–11.
[22] Yuan M, Wu Y, Lin L. Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network. In: Proceedings of IEEE International Conference on Aircraft Utility Systems; 2016. p. 135–40.
[23] Zhao R, Wang J, Yan R, Mao K. Machine health monitoring with LSTM networks. In: Proceedings of 10th International Conference on Sensing Technology; 2016. p. 1–6.
[24] Hinton GE, Salakhutdinov RR. Reducing the dimensionality of data with neural networks. Science 2006;313(5786):504.
[25] Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. In: Proceedings of 26th Annual Conference on Neural Information Processing Systems, 2; 2012. p. 1097–105.
[26] Hinton GE, Deng L, Yu D, Dahl G, Mohamed AR, Jaitly N, et al. Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. IEEE Signal Process Mag 2012;29(6):82–97.
[27] Ren L, Cui J, Sun Y, Cheng X. Multi-bearing remaining useful life collaborative prediction: a deep learning approach. J Manuf Syst 2017;43, Part 2:248–56.
[28] Liao L, Jin W, Pavel R. Enhanced restricted boltzmann machine with prognosability regularization for prognostics and health assessment. IEEE Trans Ind Electron 2016;63(11):7076–83.
[29] Zhang C, Lim P, Qin AK, Tan KC. Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics. IEEE Trans Neural Netw Learn Syst 2016;PP(99):1–13.
[30] Saxena A, Goebel K, Simon D, Eklund N. Damage propagation modeling for aircraft engine run-to-failure simulation. In: Proceedings of International Conference on Prognostics and Health Management; 2008. p. 1–9.
[31] Babu GS, Zhao P, Li X-L. Deep convolutional neural network based regression approach for estimation of remaining useful life. In: Database systems for advanced applications: 21st international conference. Cham: Springer International Publishing; 2016. p. 214–28.
[32] Xu J, Wang Y, Xu L. PHM-oriented integrated fusion prognostics for aircraft engines based on sensor data. IEEE Sensors J 2014;14(4):1124–32.
[33] Abdel-Hamid O, Mohamed A, Hui J, Penn G. Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition. In: Proceedings of IEEE international conference on acoustics, speech and signal processing; 2012. p. 4277–80.
[34] Szegedy C, Liu W, Yangqing J, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: Proceedings of IEEE conference on computer vision and pattern recognition; 2015. p. 1–9.
[35] Ren W, Liu S, Zhang H, Pan J, Cao X, Yang M-H. Single image dehazing via multi-scale convolutional neural networks. In: European conference on computer vision. Cham: Springer International Publishing; 2016. p. 154–69.
[36] Sun W, Shao S, Zhao R, Yan R, Zhang X, Chen X. A sparse auto-encoder-based deep neural network approach for induction motor faults classification. Measurement 2016;89:171–8.
[37] Liu B, Liu J, Bai X, Lu H. Regularized hierarchical feature learning with non-negative sparsity and selectivity for image classification. In: Proceedings of 22nd Int Conf Pattern Recognit; 2014. p. 4293–8.
[38] Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. J Mach Learn Res 2010;9:249–56.
[39] Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. Nature 1986;323(6088):533–6.
[40] Kingma D., Ba J.. Adam: a method for stochastic optimization. arXiv preprint arXiv:14126980 2014;.
[41] Saxena A, Goebel K. Turbofan engine degradation simulation data set. NASA Ames Prognostics Data Repository. Moffett Field, CA: NASA Ames Research Center; 2008. (http://tiarcnasagov/project/prognostic-data-repository)
[42] Ramasso E. Investigating computational geometry for failure prognostics. Int J Prognostics Health Manage 2014;5(1):005.
[43] Coble JB, Hines JW. Prognostic algorithm categorization with PHM challenge application. In: Proceedings of international conference on prognostics and health management; 2008. p. 1–11.
[44] Wang P, Youn BD, Hu C. A generic probabilistic framework for structural health prognostics and uncertainty management. Mech Syst Signal Process 2012;28:622–37.
[45] Guo X, Chen L, Shen C. Hierarchical adaptive deep convolution neural network and its application to bearing fault diagnosis. Measurement 2016;93:490–502.

[46] Guo L, Li N, Jia F, Lei Y, Lin J. A recurrent neural network based health indicator for remaining useful life prediction of bearings. Neurocomputing 2017;240:98–109.
[47] Malhotra P., TV V., Ramakrishnan A., Anand G., Vig L., Agarwal P., et al. Multi-sensor prognostics using an unsupervised health index based on LSTM encoder-decoder. arXiv preprint arXiv:160806154 2016;.
[48] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of 32nd International Conference on Machine Learning, 1; 2015. p. 448–56. Lile, France
[49] Peng Y, Wang H, Wang J, Liu D, Peng X. A modified echo state network based remaining useful life estimation approach. In: Proceedings of IEEE Conference on Prognostics and Health Management; 2012. p. 1–7.
[50] Louen C, Ding SX, Kandler C. A new framework for remaining useful life estimation using support vector machine classifier. In: Proceedings of IEEE Conference on Control and Fault-Tolerant Systems; 2013. p. 228–33.
[51] Lim P, Goh CK, Tan KC. A time window neural network based framework for remaining useful life estimation. In: Proceedings of International Joint Conference on Neural Networks; 2016. p. 1746–53.