Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 217 (2023) 456–464

Procedia Computer Science

www.elsevier.com/locate/procedia

4th International Conference on Industry 4.0 and Smart Manufacturing

Self-Attention Transformer-Based Architecture for Remaining Useful Life Estimation of Complex Machines

Abdul Wahid[a,*], Muhammad Yahya[a], John G Breslin[a], Muhammad Ali Intizar[b]

[a]*Data Science Institute (DSI), National University of Ireland Galway, Galway Ireland*
[b]*Dublin City University, Glasnevin, Dublin 9,*

Abstract

Meaningful feature extraction from multivariate time-series data is still challenging since it takes into account the correlation between pairs of sensors as well as the temporal information of each time-series. Meanwhile, the huge industrial system has evolved into a data-rich environment, resulting in the rapid development and deployment of deep learning for machine RUL prediction. RUL (Remaining Useful Life) examines a system's behavior over the course of its lifetime, that is, from the last inspection to when the system's performance deteriorates beyond a certain point. RUL has been addressed using Long-Short-Term Memory (LSTM) and Convolution Neural Network (CNN), particularly in complex tasks involving high-dimensional nonlinear data. The main focus, however, has been on degradation data. In 2021, a new realistic run-to-failure turbofan engine degradation dataset was released, which differs significantly from the simulation dataset. The key difference is that each cycle's flight duration varies, so the existing deep technique will be ineffective at predicting the RUL for real-world degradation data. We present a Self-Attention Transformer-Based Encoder model to address this problem. The encoder with the time-stamp encoder layer works in parallel to extract features from various sensors at various time stamps. Self-attention enables efficient processing of extended sequences and focuses on key elements of the input time series. Self-attention is used in the proposed Transformer model to access global characteristics from diverse time-series representations. Under real-world flight conditions, we conduct tests on turbofan engine degradation data using variable-length input. The proposed approach for estimating RUL of turbofan engines appears to be efficient based on empirical results.

Keywords: Remaining Useful Life (RUL); Deep Learning (DL); Self-Attention (SA); Transformer Model (TM)

1. Introduction

For reliable, efficient and successful operation of modern complex mechanical equipment maintenance and prognostic health management (PhM) are of crucial importance. Prediction of Remaining Useful Life (RUL) plays an

[*] Abdul Wahid. Tel.: +353-830-89-2514
E-mail address: a.wahid2@nuilgalway.ie

1877-0509 © 2022 The Authors. Published by Elsevier B.V.

10.1016/j.procs.2022.12.241

important role in the predictive maintenance of large equipment's. With the goal of estimating the performance of machinery over it's lifetime period and providing a suitable maintenance plan to avoid serious accidents [2]. The rise of internet of things (IoT) and industrial digitization has gradually transformed present industrial systems into a data-rich environment. These developments have created an unprecedented opportunities to study and develop advanced methods to predict the Remaining Useful Life (RUL) of complex machines using machine and deep learning techniques [1].

Traditional model-based, data-driven, and hybrid methods are the three types of RUL prediction methods. To characterize the degradation trend of components, the model-based method necessitates accurate dynamic modeling of mechanical equipment or components [3]. Moreover, modern industrial large-scale equipment, on the other hand, is becoming increasingly complicated, with a variety of nonlinear connections between diverse systems. As a result, establishing a precise model is challenging. The purpose of the data-driven RUL prediction method is to find a mapping link between RUL and target equipment attributes [4]. However for complex mechanical equipment an expert knowledge and physical modelling is not required [5]. Recently many studies have been proposed for predicting the RUL prediction methods based on deep learning architectures [6] [8]. However a key issue for RUL prediction is that more robust methods are required which can extract the features that contain more and useful degradation information. According to Qin et al [9] attention mechanism is capable of learning different dependencies across multiple sensors at different time-stamps. Recently many studies have been proposed that attempted to predict RUL by combining the attention mechanism with CNN and LSTM structures [8] [10]. However these methods have two major problems. First, the LSTMS and CNN'S fails to capture the long-term dependencies efficiently and secondly the attention mechanism in LSTM and CNN suffers from collective influence between the extracted features because the data is fed sequentially into the network, therefore affecting the RUL prediction.

Recently Transformer architecture [11] were introduced in sequence modelling to process variable input length. It uses the self-attention mechanism to extract features and captures the long-term dependencies between components in a sequence without taking into account their distance. And as a result they remain less affected despite the increase in the sequence length as compared to LSTM and CNN. Our research aims to address the above mentioned challenges by proposing a transformer based deep learning method to predict the RUL of turbofan engine. The main contributions of this research can be summarized as follows:

- We propose a self-attention based transformer architecture for predicting the RUL. We use NASA's 2021 run-to-failure turbofan engine dataset for experimentation.
- We investigate the abstract feature extraction layer module to incorporate more important features from variable-length time-stamps without any prior domain knowledge into the attention mechanism.

The rest of the paper is organized as follows. Section 2 introduces related works. Section 3 illustrates the proposed methodology. Section 4 and 5 demonstrates the effectiveness of the proposed approach and discuss the results. Finally, section 6 concludes the paper.

2. Related Works

RUL estimation for turbofan engines has attracted a lot of research interest due to the importance of its application. However, many researchers have investigated the advanced method and have primarily relied on simulation datasets because practically all practical turbofan engine data is extremely valuable and even proprietary. Previously NASA in 2008 published a turbofan engine dataset knows as C-MAPSS dataset [13]. In 2021 a new run-to-failure turbofan engine degradation dataset (N-CMAPSS) [12] was published by the Prognostics CoE at NASA in collaboration with ETH Zurich and PARC. It delivers actual flight data of turbofan engines under real settings ranging from both normal and fault operation settings as compared to the former dataset.

In literature RNN based architectures and it's variants such as LSTM and GRU have been widely used for RUL predictions. Heimes et al [14] used RNN for RUL prediction. The LSTM prediction model for RUL estimation was used by [15]. A GRU based method for remaining useful life prediction of nonlinear deterioration process was proposed in [16]. Apart from RNN and LSTM, CNN has also been applied for RUL prediction tasks. A deep CNN was proposed in [17] which used data normalization and performed convolution across time dimension. In [18] multi-scale CNN was proposed for RUL estimation, it focused on keeping the global and local information in balance in comparison to normal CNN's. Furthermore, many studies leveraged the combination of CNN and LSTM [19], other studies used sequence learning and attention based architectures [20] [8] for RUL estimation.

Transformer architecture has recently been used in time-series related jobs due to its effectiveness in modeling extended sequences. Zhou et al. [21] investigated the use of Transformer in the prediction of long sequence time series and presented the ProbSparse self-attention technique to reduce time complexity and memory utilization. The Longformer was proposed by [22], and it has an attention mechanism that grows linearly with sequence length, making it easier to digest large sequences. Currently only few studies regarding the use of transformer architecture for RUL estimation are available. In this research, we investigate this direction and present a self-attention based transformer architecture capable of simultaneously capturing the weight information of various sensors at different time-stamps and hence improve the overall RUL estimation.

3. Proposed Approach

This section illustrates the proposed approach in detail as shown in Figure 1. The proposed model has been inspired by Transformer architecture in natural language processing (NLP). The model consists of three substructures self-attention layer, encoder layer, and prediction layer. Unlike other RUL prediction methods based on RNN and CNN methods, the proposed model uses a self-attention mechanism to capture long-term dependence information between sequence inputs and outputs without taking distance into account, so the importance of each work cycle information is not diminished as time step length increases. We adopt abstract feature extraction strategy based on the Transformer architecture that is better suited for RUL prediction.

3.1. Transformer Architecture

Transformer architecture was first introduced in 2017 by Vaswani et al [11]. It's a sequence-to-sequence based encoder decoder architecture. The encoder converts the input sequence into a higher-dimensional vector, which is then fed into the decoder, which produces an output sequence. Transformer, unlike RNNs, develops long-term dependencies via a dot-product-based attention mechanism. Many natural language processing tasks, such as machine translation [11], named entity recognition, general language understanding, and question answering [23], have achieved improved results with transformers.

This research proposes a self-attention based transformer architecture for RUL estimation, to address the limitations of RNN and CNN based methods as explained in section 1. Furthermore, we adopt the self-attention model which is a normal attention model. It generates the features from the same item of the sequential input and models the sequential data by adding it to the prior input sequence. And as a result sensitivity of the model to local information is increased.

3.2. Encoder layer

We use transformer encoder layer to capture the long-term and short-term dependencies for RUL estimation. The encoder layer used in this architecture is based on [24]. It is composed on N multiple sensor encoder layers, time-stamp layers and input embedding layers. In order to prepare for the feature extraction process, the input embedding layer transfers the input state monitoring data to a vector using a feed forward network (FFN). A multi-head sensor self-attention layer and an FFN layer are the two primary sub-layers of a sensor encoder layer. After each sub-layer, there is a residual connection and normalization layer. The goal of residual connection is to make training a deep neural network easier. The sensor encoder layer and the time step encoder layer have the same structure. A multi-head time step self-attention layer and an FFN layer are the two major sub-layers. The distinction is that the time step encoder layer captures features along the time step dimension, allowing the model to focus on the time steps that matter most for RUL prediction. We give a brief overview of each module here, please refer to [24] for more details.

3.3. Attention blocks

Transformer architecture uses multi-head attention to access the features in the encoder and decoder based on the sequence to sequence model, rather than CNN or RNN Networks. Word embedding is used to process the input

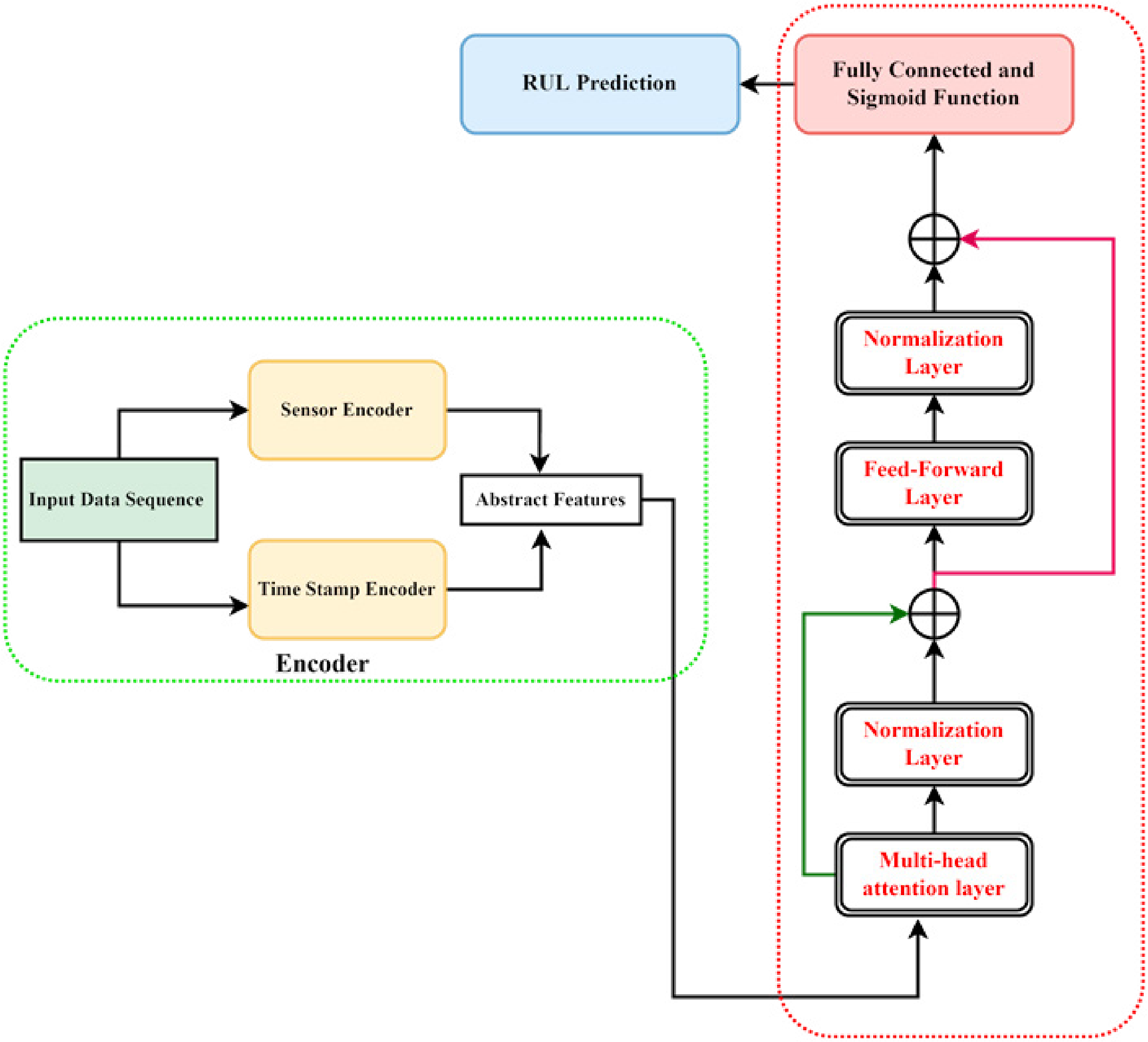

Fig. 1. Overall architecture of the proposed model.

sequence, as well as positional embedding, which introduces the positional link between elements. Self-attention captures the global and simultaneous dependencies between all parts. As a result, during the training phase, parallel computation becomes possible.

3.3.1. Multi-head attention

Multi-head attention is actually composed of several heads of self-attention. A mapping and query is established by each head of the attention modules from the key-value pairs to the output by an attention function where the query, keys, and values are all matrices made up of vectors transformed from the previous output. It’s worth to note that self-attention mechanism i used here. The result is a weighted sum of the values, with the weights allocated to each value determined by the query’s compatibility function with the relevant key. An attention function on a given set of query vectors and key-value pairs is calculated as follows:

$$K_{\mathrm{j}} = fW_j^{\,k} \tag{1}$$

$$V_{\mathrm{j}} = fW_j^{\,v} \tag{2}$$

$$Q_{\mathrm{j}} = fW_j^{\,q} \tag{3}$$

Where $f\,W_j^k, f\,W_j^v, f\,W_j^q$ are trainable weighted matrix. The derived output from each module is the weighted sum of the values. A scaled dot-product attention is applied on the derived weights K_{j}, V_{j}, and Q_{j}.

$$t^{j}{}_{attention} = Softmax\left(\frac{Q_{\mathrm{j}}K_{\mathrm{j}}{}^{\mathrm{T}}}{\sqrt{d_{\mathrm{k}}}}\right)V_{\mathrm{j}} \tag{4}$$

Furthermore, we concatenate the information from multiple sub-spaces at different locations. The multi-head attention can be formulated as follows:

$$h_{\mathrm{mh}} = Concat\left(h_{\mathrm{j}}{}^{1}, h_{\mathrm{j}}{}^{2}, \ldots, h_{\mathrm{j}}{}^{\mathrm{H}}\right)W^{\mathrm{o}} \tag{5}$$

where W^{o} is also a matrix.

3.4. Abstract feature extraction

We proposed a local abstract feature extraction layer to map raw sensor data into distributed semantic representations and provide information regarding the local features to the upper layers at each time step, also taking into account the neighboring time steps in a time sequence that may have stronger dependencies. This layer extracts the important features from the sensor encoder and time-stamp encoder. The resulting features are combined together into a new feature map called abstract feature map. The sensor encoder and time-stamp encoder contain sub-layers of senor or time-stamp layers. This layer extracts features from both the encoders at the same-time as they are designed in a parallel fashion. This helps to avoid the mutual influence of information which improves the overall RUL estimation. Simultaneously the self-attention mechanism captures the long-term dependency information.

3.5. Feed forwards network (FFN)

The FFN involves two linear transformations with a ReLU activation function. A residual connection is applied to increase the convergence before the features are transmitted through FFN. It can be formulated as follows:

$$FFN(x) = ReLU(W_1 x + b_1) + W_2 + b_2 \tag{6}$$

3.6. Regression layer

Finally we implement the regression layer which gets input from the attention blocks. The regression layer r_{p} is represents the output of the encoder with respect to the input data sequence. The RUL is estimated as follows

$$g_\mathrm{t} = \sigma(\omega_\mathrm{o} r_\mathrm{p} + b_\mathrm{o}) \tag{7}$$

where g_t is the estimated RUL σ denotes the sigmoid activation function and ω and b_o are the scalar objects. The model uses mean square error to calculate the model loss and min-max normalization. The loss function is calculated as follows:

$$MSE = \frac{\sum^{\mathrm{N}}{}_{\mathrm{i}=1}(\hat{r} - r_\mathrm{i})^2}{N} \tag{8}$$

4. Experiments

4.1. N-CMAPSS Datatset

The N-CMAPSS dataset [12] simulates the run-to-failure deterioration trajectories of a fleet of turbofan engines with uncertain initial health states under real-world flight conditions. The N-CMAPSS dataset currently contains eight subsets of data from 128 units, as well as seven possible failure scenarios that affect the flow (F) and/or efficiency (E) of all rotating sub-components. The overall useful life has been set to 100%. The label for each flight cycle is calculated by dividing the current cycle's index by the unit's total number of cycles. The label is a positive decimal between 0 and 1 in this fashion. The higher the number, the more cycles the engine will be able to support. Table 2 gives the overview of the dataset.

We consider DS03 dataset for our experiment. As shown in Table 2, DS03 has 9.8 million rows, 15 Units (i.e. data for 15 different turbofan engines), 3 flight classes and 1 failure modes. The input to the model is the data from the measurements, virtual sensors, and model health parameters. Data for multiple modalities may be missing due to a diversity of flight conditions and the inequality of flight length during each flight cycle. The data set is partitioned in the following fashion during the training phase. 70% of the data is used in training, 10% for testing and the remaining 10% is used for validation.

Table 1. A detailed overview of the N-CMAPSS dataset.

Name	Units	Flight Classes	Failure Modes	Size
DS01	10	1,2,3	1	7.6 M
DS02	9	1,2,3	2	6.5 M
DS03	15	1,2,3	1	9.8 M
DS04	10	2,3	1	10.0 M
DS05	10	1,2,3	1	6.9 M
DS06	10	1,2,3	1	6.8 M
DS07	10	1,2,3	1	7.2 M
DS08	54	1,2,3	1	35.6 M

4.1.1. Implementation details

The proposed approach is implemented using PyTorch 1.3.2. The model is trained for 800 epochs. The RUL labels for run-to failure dataset are normalized between [0, 1]. We used 6 attention blocks and 16 heads of self-attentions with a dropout ratio of 0.1 according to [25]. The performance of the proposed model for RUL estimation is evaluated using root mean square error (RMSE) and mean absolute error [2]. The following two metrics are given as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |\hat{x}_{\mathrm{i}} - y_{\mathrm{i}}| \tag{9}$$

$$RMSE = \frac{1}{N}\sum_{i=1}^{N}(\hat{x}_{\mathrm{i}} - y_{\mathrm{i}})^2 \tag{10}$$

5. Results and discussion

The results of the N-CMPASS dataset are shown in Figure **??**. The anticipated remaining useful life results are shown by the blue curves, while the ground truth labels are represented by the black dashed lines. Except for a few specific flight cycles, the predicted spots are slightly fluctuant within a limited range, demonstrating the effectiveness of the proposed model. We used two metrics to evaluate the performance quantitatively. At different epochs, MAE, RMSE, [2] provide slightly different best models. As seen in Table 2, the RMSE values of all the units are quite close to 1, indicating that the forecasts and ground truth are highly comparable. When it comes to MAE, and RMSE, the lower these measures are, the better the regression performance. The MAE, and RMSE values are all close to 0, indicating a high degree of similarity between the predictions and the labels. Units with short length cycles, in contrast to those with longer flight lengths, have bigger prediction errors. Empirical results from real-world flying scenarios show that the length of each cycle has a direct impact on models performance. As a result, units with shorter cycles lack the necessary information for prediction, resulting in extreme sparsity.

Table 2. Comparison of MAE and RMSE for N-CMAPSS dataset..

Unit	2	5	10	16	18	20
Loss	5.10e-3	5.60e-3	6.73 e-3	3.10e-3	2.02e-3	6.10e-3
MAE	4.16e-2	3.90e-2	4.01e-2	3.11e-2	3.97e-2	4.02e-2
RMSE	6.19e-2	6.81e-2	7.51e-2	4.5e-2	8.46e-2	8.97e-2

6. Conclusion

This paper proposes a transformer based architecture for RUL estimation, capable of capturing both short-term and long-term dependencies a in given time sequence. In contrast to methods based on CNN, our model is built on the a dot-product self-attention mechanism over all time steps. To capture the properties of degradation data throughout flight cycles under real-world flight conditions, our proposed model employs a multi-head self-attention mechanism. The abstract features provide access to more significant degradation features that aid in RUL estimation. Under real-world flight conditions, the model can estimate the RUL of turbofan engine degradation. The suggested model with variable-length input is effective and robust, as evidenced by experimental results and analysis. Furthermore, because of the sufficiency of the presented information, the length of flight cycles has a direct impact on accuracy.

References

[1] Lei Y, Li N, Guo L, Li N, Yan T, Lin J. Machinery health prognostics: A systematic review from data acquisition to RUL prediction. Mechanical systems and signal processing. 2018 May 1;104:799-834.

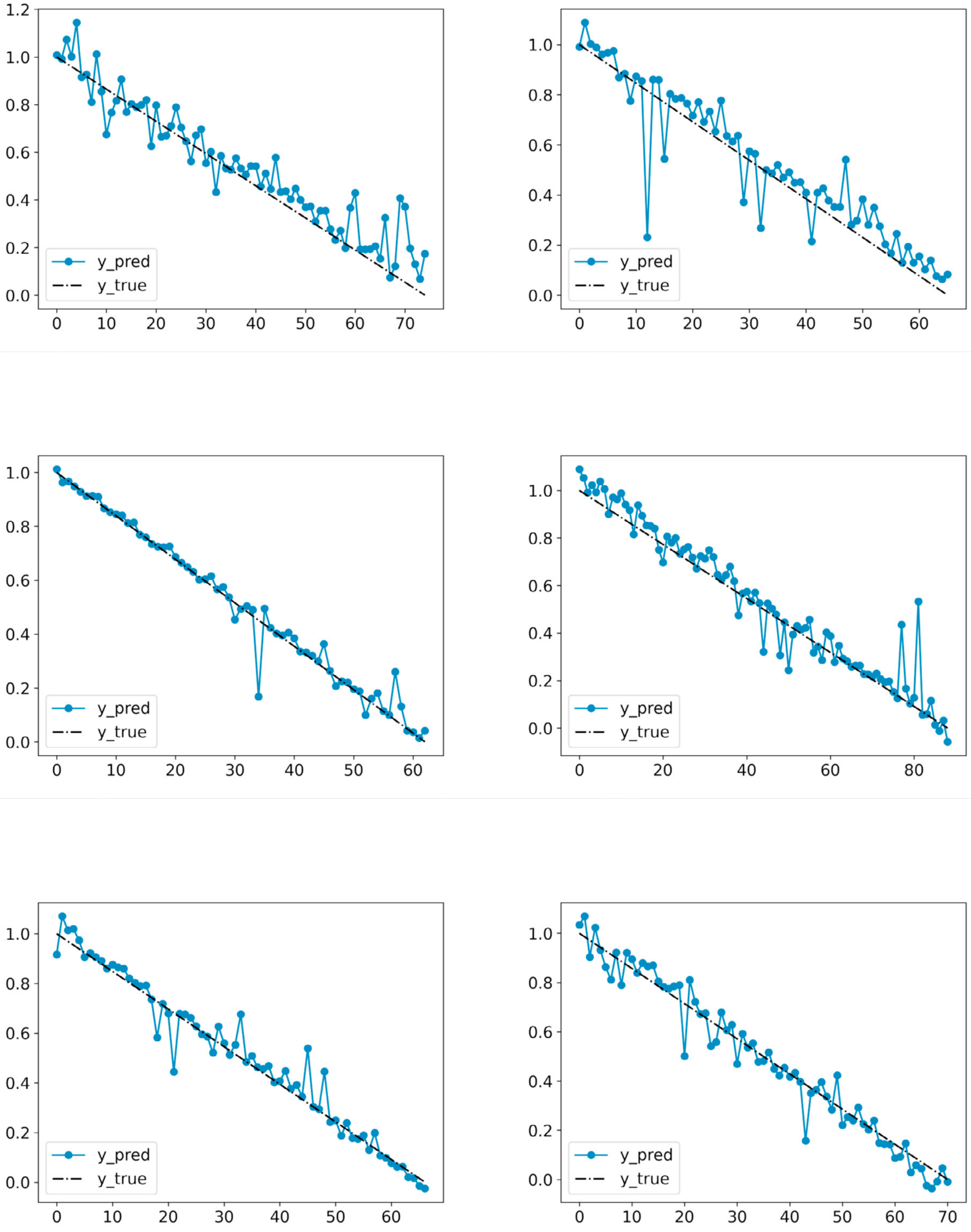

Fig. 2. RUL estimation results for the N-CMAPSS dataset. The results represent engines 2, 5, 10, 16, 18, and 20. The predicted estimation is represented in blue and true estimations are represented in black.

[2] Zhang C, Lim P, Qin AK, Tan KC. Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics. IEEE transactions on neural networks and learning systems. 2016 Jul 11;28(10):2306-18.

[3] Park J, Ha JM, Oh H, Youn BD, Choi JH, Kim NH. Model-based fault diagnosis of a planetary gear: A novel approach using transmission error. IEEE Transactions on Reliability. 2016 Aug 30;65(4):1830-41.

[4] Qin Y, Chen D, Xiang S, Zhu C. Gated dual attention unit neural networks for remaining useful life prediction of rolling bearings. IEEE Transactions on Industrial Informatics. 2020 Jun 2;17(9):6438-47.

[5] Meng H, Li YF. A review on prognostics and health management (PHM) methods of lithium-ion batteries. Renewable and Sustainable Energy Reviews. 2019 Dec 1;116:109405.

[6] Xia M, Zheng X, Imran M, Shoaib M. Data-driven prognosis method using hybrid deep recurrent neural network. Applied Soft Computing. 2020 Aug 1;93:106351.

[7] Wahid A, Breslin JG, Intizar MA. Prediction of Machine Failure in Industry 4.0: A Hybrid CNN-LSTM Framework. Applied Sciences. 2022 Apr 22;12(9):4221.

[8] Chen Z, Wu M, Zhao R, Guretno F, Yan R, Li X. Machine remaining useful life prediction via an attention-based deep learning approach. IEEE Transactions on Industrial Electronics. 2020 Feb 13;68(3):2521-31.

[9] Qin Y, Cai N, Gao C, Zhang Y, Cheng Y, Chen X. Remaining Useful Life Prediction Using Temporal Deep Degradation Network for Complex Machinery with Attention-based Feature Extraction. arXiv preprint arXiv:2202.10916. 2022 Feb 21.

[10] Song Y, Gao S, Li Y, Jia L, Li Q, Pang F. Distributed attention-based temporal convolutional network for remaining useful life prediction. IEEE Internet of Things Journal. 2020 Jun 23;8(12):9594-602.

[11] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. Advances in neural information processing systems. 2017;30.

[12] Arias Chao M, Kulkarni C, Goebel K, Fink O. Aircraft engine run-to-failure dataset under real flight conditions for prognostics and diagnostics. Data. 2021 Jan 13;6(1):5.

[13] Saxena A, Goebel K, Simon D, Eklund N. Damage propagation modeling for aircraft engine run-to-failure simulation. In2008 international conference on prognostics and health management 2008 Oct 6 (pp. 1-9). IEEE.

[14] Heimes FO. Recurrent neural networks for remaining useful life estimation. In2008 international conference on prognostics and health management 2008 Oct 6 (pp. 1-6). IEEE.

[15] Cheng Y, Wu J, Zhu H, Or SW, Shao X. Remaining useful life prognosis based on ensemble long short-term memory neural network. IEEE Transactions on Instrumentation and Measurement. 2020 Oct 15;70:1-2.

[16] Chen J, Jing H, Chang Y, Liu Q. Gated recurrent unit based recurrent neural network for remaining useful life prediction of nonlinear deterioration process. Reliability Engineering & System Safety. 2019 May 1;185:372-82.

[17] Li X, Ding Q, Sun JQ. Remaining useful life estimation in prognostics using deep convolution neural networks. Reliability Engineering & System Safety. 2018 Apr 1;172:1-1.

[18] Zhu J, Chen N, Peng W. Estimation of bearing remaining useful life based on multiscale convolutional neural network. IEEE Transactions on Industrial Electronics. 2018 Jun 13;66(4):3208-16.

[19] Li J, Li X, He D. A directed acyclic graph network combined with CNN and LSTM for remaining useful life prediction. IEEE Access. 2019 May 28;7:75464-75.

[20] Yang H, Ding K, Qiu RC, Mi T. Remaining useful life prediction based on normalizing flow embedded sequence-to-sequence learning. IEEE Transactions on Reliability. 2020 Aug 6;70(4):1342-54.

[21] Zhou H, Zhang S, Peng J, Zhang S, Li J, Xiong H, Zhang W. Informer: Beyond efficient transformer for long sequence time-series forecasting. InProceedings of the AAAI Conference on Artificial Intelligence 2021 May 18 (Vol. 35, No. 12, pp. 11106-11115).

[22] Beltagy I, Peters ME, Cohan A. Longformer: The long-document transformer. arXiv preprint arXiv:2004.05150. 2020 Apr 10.

[23] Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. 2018 Oct 11.

[24] Zhang Z, Song W, Li Q. Dual-Aspect Self-Attention Based on Transformer for Remaining Useful Life Prediction. IEEE Transactions on Instrumentation and Measurement. 2022 Mar 17;71:1-1.

[25] Ma Q, Zhang M, Xu Y, Song J, Zhang T. Remaining Useful Life Estimation for Turbofan Engine with Transformer-based Deep Architecture. In2021 26th International Conference on Automation and Computing (ICAC) 2021 Sep 2 (pp. 1-6). IEEE.