University of Reading

Predictive maintenance using cox proportional hazard deep learning

Article

Accepted Version

Creative Commons: Attribution-Noncommercial-No Derivative Works 4.0

Chen, C., Liu, Y., Wang, S. ORCID: https://orcid.org/0000-0003-2113-5521, Sun, X., Di Cairano-Gilfedder, C., Titmus, S. and Syntetos, A. A. (2020) Predictive maintenance using cox proportional hazard deep learning. Advanced Engineering Informatics, 44. 101054. ISSN 1474-0346 doi: 10.1016/j.aei.2020.101054 Available at https://centaur.reading.ac.uk/89373/

It is advisable to refer to the publisher's version if you intend to cite from the work. See Guidance on citing.

To link to this article DOI: http://dx.doi.org/10.1016/j.aei.2020.101054

Publisher: Elsevier

All outputs in CentAUR are protected by Intellectual Property Rights law, including copyright law. Copyright and IPR is retained by the creators or other copyright holders. Terms and conditions for use of this material are defined in the End User Agreement.

www.reading.ac.uk/centaur

University of Reading

CentAUR

Central Archive at the University of Reading

Reading's research outputs online

Predictive Maintenance Using Cox Proportional Hazard Deep Learning

Chong Chen[a], Ying Liu[a*], Shixuan Wang[b, c], Xianfang Sun[d], Carla Di Cairano-Gilfedder[e], Scott Titmus[f] and Aris A. Syntetos[c]

[a] *Institute of Mechanical and Manufacturing Engineering, School of Engineering, Cardiff University, Cardiff, CF24 3AA, UK*

[b] *Department of Economics, University of Reading, Reading RG6 6AA, United Kingdom*

[c] *Panalpina Centre for Manufacturing and Logistics Research, Cardiff Business School, Cardiff University, Cardiff, CF10 3EU, UK*

[d] *School of Computer Science and Informatics, Cardiff University, Queen's Buildings, Cardiff CF24 3AA, UK*

[e] *Applied Research, BT Technology, Ipswich, IP5 3RE, UK*

[f] *BT, Solihull, B37 7YN, UK*

[*] *Corresponding author. E-mail: LiuY81@Cardiff.ac.uk*

Abstract

Predictive maintenance (PdM) has become prevalent in the industry in order to reduce maintenance cost and to achieve sustainable operational management. The core of PdM is to predict the next failure so corresponding maintenance can be scheduled before it happens. The purpose of this study is to establish a Time-Between-Failure (TBF) prediction model through a data-driven approach. For PdM, data sparsity is regarded as a critical issue which can jeopardize algorithm performance for the modelling based on maintenance data. Meanwhile, data censoring has imposed another challenge for handling maintenance data because the censored data is only partially labelled. Furthermore, data sparsity may hurt algorithm performance of the existing approaches when addressing the data censoring issue. In this study, a new approach called Cox proportional hazard deep learning (CoxPHDL) is proposed to tackle the aforementioned issues of data sparsity and data censoring that are common in the analysis of operational maintenance data. The idea is to offer an integrated solution by taking advantage of deep learning and reliability analysis. To start with, an autoencoder is adopted to convert the nominal data into a robust representation. Secondly, a Cox PHM is researched to estimate the TBF of the censored data. A long-short-term memory (LSTM) network is then established to train the TBF prediction model based on the pre-processed maintenance data. Experimental studies using a sizable real-world fleet maintenance data set provided by a UK fleet company have demonstrated the merits of the proposed approach where the algorithm performance based on the proposed LSTM network has been improved respectively in terms of MCC and RMSE.

Keywords: Predictive Maintenance; Deep Learning; Reliability Analysis; Fleet Maintenance; Machine Learning

1. Introduction

Maintenance is critical as it is highly relevant to equipment lifespan. The useful life of a system can be extended with the implementation of maintenance. With an accurate prediction of equipment failure time, maintenance can be scheduled beforehand so to decrease the probability of accidents, economic losses, and human casualty. Nowadays, predictive maintenance (PdM) has been widely applied in different industries such as automobile [1], aircraft [2], manufacturing [3], etc. The equipment's pending failures can be detected and failure time can be predicted in advance using data analytic tools such as defined health factors, statistical inference methods, and engineering approaches [4]. In PdM, it is essential to predict the next failure time accurately. If maintenance is implemented too early in advance, the benefits of more extended usage are lost. In contrast, if it is carried out too late, the equipment may fail and result in a larger loss [5]. Hence, improving the prediction accuracy of the next failure can bring tangible benefits to the industry.

Recently, data-driven approaches have been widely explored in PdM. Among these studies, condition-based PdM has gained increasing attention. With the development of the Internet of Things, an increasing amount of data relevant to the health condition of equipment can be collected with the aim to implement PdM [6]. Real-time sensor data offers dynamic information of part's condition, which can be used for the remaining useful life prediction [7, 8]. However, the deployment of sensors to collect real-time data requires extra cost, especially for a fleet management company which possess numerous automobiles. In strike contrast, historical maintenance data is relatively easy to obtain. Different from the studies that explored RUL modelling based on sensor data, the TBF modelling also can be conducted based on historical maintenance data. An accurate TBF prediction can be helpful to a fleet management company to optimise its management. Normally, there are both nominal data and numeric data in historical maintenance dataset. The nominal data, such as model type and geographical location, are of significance to the equipment failure time. Traditionally, the nominal data is encoded into binary attributes using one-hot encoding, which may considerably increase the dimension and sparsity of the

dataset [9]. When the dimension of binary attribute significantly exceeds the dimension of numeric data, the dataset becomes sparse and therefore algorithms may not able to learn the hidden patterns [10]. Hence, it is worthwhile to develop a more appropriate method to represent the nominal data. Autoencoder, as a deep learning algorithm, has shown its advantages in extracting robust features and reducing dimension [11, 12]. Hence, it can be a helpful tool in data pre-processing for historical maintenance data.

Traditionally, the data analytics of historical maintenance data has strongly relied on Cox PHM and its variants which is used to learn the relationship between reliability and exogenous variables due to its suitability for processing uncensored and censored data [13-15]. The most common type of censoring in automobile maintenance is right-censoring. The corrective maintenance is executed when an automobile fails, and the actual failing time is recorded. However, the preventive maintenance is carried out when an automobile is deemed to be failed soon, and therefore the actual failing time is semi-known (only the minimum of TBF is known). To be specific, the data of corrective maintenance is uncensored, and the data on preventive maintenance is right-censored. The relationship between equipment lifespan and reliability can be modelled by Cox PHM. Moreover, the data sparsity might impact the performance of Cox PHM. Therefore, robust features (covariates) are needed to generate a Cox PHM, which has a high requirement in feature selection [16].

Recently, machine learning, as a subset of artificial intelligence, has been widely used in different areas of industry such as energy consumption prediction [17], fault diagnosis [18], and adaptive control optimisation [19]. Meanwhile, it also has been used in PdM [1, 20-27]. Among various machine learning methods, deep learning has gained considerable attention in PdM [23-27]. Deep learning, as a group of machine learning techniques, has shown its merits in modelling based on high dimensional and large size data [28]. LSTM network, as one of the deep learning techniques, has a specialized structure in processing sequential data [29]. Due to the next TBF of an automobile is highly relevant to its previous

failure and maintenance information, LSTM network can be employed in the modelling of automobile TBF.

The algorithm performance of deep learning relies on the quality of the data label [30]. Without an accurate label, deep learning may be challenging to learn the hidden patterns within data. In historical maintenance data, the label of the censored data is not sufficiently accurate. Hence, the data censoring problem needs to be addressed. This study aims to propose an approach called Cox proportional hazard deep learning (CoxPHDL) to build a TBF prediction model based on historical maintenance data. The main contributions of this paper are: (1) Different from most of the existing studies aiming at RUL modelling based on sensor data, an automobile TBF modelling approach based on historical maintenance data is proposed in this study; (2) Due to the data sparsity might damage the algorithm performance of LSTM network and Cox PHM, autoencoder is introduced to convert the sparse data into robust representation; (3) Because accurate data label is important for deep learning modelling, Cox PHM is introduced to estimate the correct label of censored data so as to improve the algorithm performance of LSTM network. The rest of this paper is organised as follows: The application of statistical approaches, machine learning approaches, and their mixing approaches in PdM are reviewed in Section 2. Section 3 introduces Cox proportional hazard deep learning. An experimental study is demonstrated in Section 4 and its results are demonstrated and discussed in Section 5. Finally, Section 7 concludes.

2. Literature Review

There are various approaches have been developed for modelling in PdM, including statistical and machine learning approaches. In statistical models, parametric models and semi-parametric models have been widely studied in PdM. Meanwhile, in recent years, researchers have introduced machine learning approaches for equipment lifecycle modelling in PdM. In the existing studies of PdM, two

issues which are data sparsity and data censoring have been widely studied using both statistical and machine learning approaches.

2.1. Statistical Approaches for Predictive Maintenance

The parametric model assumes time-to-failure or TBF follows a specific statistical distribution such as Weibull [31]. The parametric models are useful when the sample follows a distribution, and the model parameters need to estimate accurately. However, in the actual cases, the algorithm performance tends to be compromised if the distribution is not properly specified. Xie and Lai (1996) [31] studied a Weibull model to estimate the lifetime distribution for electrical and mechanical components. The model is based on two Weibull survival functions, and a graphical estimation approach was adopted to estimate the parameters for the Weibull model. Mettas (2000) [32] proposed a versatile accelerate failure time model to investigate the accelerate life data. The algorithm combines the life-stress relationships for one or two types of stresses with a model formulated by different distribution such as Weibull and Lognormal. The proposed model can be used to generate the relationship of product life with multi-stresses-types, while traditional AFT models can only generate the relationship of product life with single stresses-type. It can be seen from both cases that the data used for modelling was in small size. The robustness of the models in actual cases needs to be further investigated.

Besides parametric models, another type of statistical model is semi-parametric. Cox PHM and its variants have been widely used in PdM due to its flexibility and ability in modelling based on uncensored and censored data [13-15]. It was used to analyse the relationship between time-independent covariates and hazard function [33]. As the standard Cox PHM is only suitable the time-independent covariates, researchers have proposed the variants of Cox PHM to consider the time-dependent covariates Anderson and Senthilselvan [13] proposed a two-steps PHM for the time-dependent coefficient. The proposed approach allows the varying covariates to be used in PHM. Conditional likelihood estimation was used to determine the regression covariates in this study. The two-steps PHM

showed better performance in a cancer mortality study compared to the performance of standard Cox PHM, while the estimation of its parameters can be further explored.

A proportional intensity model based on the Cox PHM called Prentice, Williams, and Peterson (PWP) was introduced based on the nonhomogeneous Poisson process to deal with recurrent failure event data. A nonhomogeneous Poisson process with power-law intensity function was adopted in this study. The proposed model can achieve better performance based on large data size and increasing failure rate [34]. Owing to the fact that Cox PHM requires robust covariates for modelling, while it may not able learn the hidden patterns from sparse covariates, Sun et al. [35] proposed a proportional covariate model (PCM) to tackle the sparsity issue in sensor data. Besides, different from the standard Cox PHM, PCM aims to predict the hazard of a system using covariates caused by the deterioration of a system, which can be used in dynamic system monitoring.

Statistical models have been investigated for decades. In most of the cases above, the datasets used in these studies are typically in small size. When the data size increases, statistical models might lack the ability to learn the hidden patterns in data due to the growing impurity and noise.

2.2. Machine Learning Approaches for Predictive Maintenance

Machine learning is a subset of artificial intelligence. It is used to learn the hidden patterns in data [36]. Recently, Machine learning model has been widely used in PdM and achieved satisfactory performance. Wei et al. (2013) [20] proposed a dynamic particle filter-support vector regression (PF-SVR) model to predict system reliability based on time series data. Parameter selection is a critical part of training a support vector regression model. In this study, a particle filter was used to learn measurement sequence of data so to estimate the parameters for SVR. Nieto et al. (2015) [21] proposed a hybrid particle swarm optimization support vector machine (PSO-SVM) to predict the RUL for aircraft engines. PSO was used to optimise the SVM kernel parameters in the model training process. PSO-SVM does not require the information previous operation stage but only use the returned engine information for modelling,

which is of advantage in the application. With the combination of the optimisation algorithm, SVM model can be more robust and applicable in PdM.

Lee and Pan (2017) [22] presented a PdM scheme for a complex system using the discrete-time Markov chain (DTMC) models and Bayesian network based on sensor data. Firstly, DTMC was used to model the degradation process of components. Then, the Bayesian network algorithm was used to model and predict the system reliability. Prytz et al. (2015) [1] proposed a data-driven approach that can predict the upcoming failures of vehicles based on the historical maintenance data and the data collected onboard the vehicles. In this approach, the random forest algorithm was used in classification modelling to identify whether the RUL of vehicles was longer or shorter than the planned interval. Since Cox PHM has its limitation in providing the business recommendation straightforwardly, Wang et al. [37] combined the Cox PHM and decision tree to a conditional inference tree to conduct reliability analysis.

In recent years, it has also been researched in PdM. Zhao et al. (2017) [23] proposed a Deep Belief Network (DBN) based method to predict the health condition of bearing in the rotating machine. DBN is a deep learning model with a hierarchical structure, and it consists of multiple stacked Restricted Boltzmann Machines. The proposed model is good at predicting the short-term health condition of bearing, and it does not rely on prognostic expertise. Li et al. (2018) [24] designed a deep convolution neural network (DCNN) to predict the RUL of aero-engine. One dimensional convolutional layer was used to extract the features from signal data. DCNN does not require expertise in prognostics and signals processing. Time window approach was adopted to validate the performance of DCNN.

Recurrent neural network (RNN) is good at processing time-series data. Malhi et al. (2011) [25] proposed a method based on competitive learning to predict long-term machine health status. The vibration data collected from rolling bearing was first pre-processed using continuous wavelet transform method. The features from raw data and the transformed data were then jointly used as the input of an RNN. As traditional RNN is poor in studying long-term patterns of data, a more advanced algorithm called LSTM network was designed to catch and store both long-term and short-term patterns of data

[29]. Yuan et al. (2016) [26] proposed a single layer LSTM network model to predict the RUL and failure probability of aero engine. Zhang et al. (2018) [27] proposed a deep LSTM network model to predict the RUL of the aero engine. The above two studies adopted the NASA C-MAPSS dataset to validate the performance of the LSTM network, and the deep LSTM network achieved better performance compared with the single LSTM network.

It can be seen from the existing studies that deep learning has been widely employed in PdM in recent years. Different from other prevailing machine learning algorithms, deep learning can identify essential features and determine model parameters automatically, and therefore, it has become prevailing in PdM.

2.3. Data sparsity and censoring in PdM

Data sparsity is a common issue in PdM which may damage the algorithm performance. It has been widely existed in studies based on sensor data. Different researchers have explored the approach to address this issue. Bastani et al. [38] proposed a sparse estimation-based classification approach to address the sparsity issue in sensor data from the real-time monitoring process. In this approach, an algorithm called greedy Bayesian method is proposed to efficiently obtain the sparse estimation from heterogeneous sensor data. Chen et al. [39] deployed sparsity-enabled signal decomposition method to establish a condition monitoring system for Automatic tool changer of CNC (Computer numerical control) machine. In this approach, morphological component analysis is used to construct the objective function for signal decomposition, which is further used to obtain the sparse representation. In the next stage, a split augmented Lagrangian shrinkage algorithm is applied to process the sparse representation to assess the operation condition. Liu et al. [40] deployed a shift-invariant sparse coding approach to extract the sparse representation from the vibration signal to conduct the bearing fault detection. In another study, sparse coding was used to transform the signature of synchronous generators to fault detection [41]. Data sparsity also exists in the study based on nominal data. Since maintenance log data consists of a large amount of nominal data, it was represented in bag-of-words format. In order to

address the data sparsity in the maintenance log dataset, a sparse linear classifier to predict the equipment failure [42].

Data censoring is another issue in PdM. Statistical approaches have been used in reliability analysis based on the censored data for decades. Jiang et al. [43] proposed a generic multivariate probabilistic macro model for reliability assessment of seepage behaviour in tunnel segments. In this study, Weibull and log-normal distributions are considered to establish a PHM for reliability estimation based on the interval censoring data. Pampuri et al. [44] proposed a PHM with l_1 penalization for semiconductor manufacturing equipment RUL prediction. Maximum Likelihood Estimation is used to obtain the covariates of PHM based on the censored data. Survival model theory is utilised with l_1 penalization to predict the RUL of equipment.

Recently, as the fast development of machine learning, statistical approaches and machine learning are jointly used in the studies of PdM. Balakrishnan et al. [45] proposed a method that using EM (Expectation-Maximization) algorithm to obtain the maximum likelihood estimates of the parameters of a two-parameter log-normal distribution, which includes the data of left truncation and right censoring. In order to address the data censoring problem in high-voltage power transformer RUL estimation, Hong et al. [46] adopted a parametric lifetime model to describe the lifetime distribution of a high-voltage power transformer to adjust its age. Based on the adjusted age, stratification and regression analysis were adopted to obtain the predictions and prediction intervals for the cumulative number of failures for the overall fleet of transformers. Random sign censoring model can be used for component failure time modelling with censored data. A Bayesian analysis of random sign censoring model is proposed to determine an optimal maintenance policy. In this study, the Bayesian approach is used to predict the next failure time [47].

In the state-of-the-art, it can be seen that deep learning has been widely deployed in the studies of PdM, while most of them have focused on RUL prediction based on sensor data. Because the historical

maintenance data is also valuable in PdM and the prediction of TBF can be helpful to a fleet management company, it is worthwhile to explore a TBF modelling approach based on historical maintenance data. Also evident from the literature is that data censoring and sparsity have been concerned in PdM. Most of the studies consider the impact of data censoring or data sparsity separately on the life cycle of equipment or system. In our previous study, we proposed a deep neural network to predict the TBF of automobile engine based on historical maintenance data. One-hot encoding approach was used to convert the nominal features. The results indicate deep learning shows merits in TBF modelling [48]. However, data sparsity and data censoring remain challenging topics in our previous study. Therefore, in this study, we proposed a new modelling approach called CoxPHDL to address both issues. For the issues of data sparsity and data censoring, the data sparsity can be a challenge when addressing the data censoring issue using statistical methods because the statistical methods have a high requirement of feature selection. Hence, a new automobile TBF modelling approach with the consideration of data sparsity and data censoring needs to be explored.

3. Method: Cox Proportional Hazard Deep Learning

A new modelling approach called CoxPHDL is proposed to establish the TBF prediction model based on the historical maintenance data, which contains both numeric and nominal features. The approach consists of three stages. The first stage is nominal data processing. Traditionally, the nominal data is converted to binary attributes for modelling. If the categories in nominal data are numerous, the dataset of binary attributes tends to be sparse. In order to lower the dimension of the sparse binary data without damaging the information dramatically, autoencoder, as a type of deep learning model which is good at extracting significant features, is introduced in this approach. The nominal data is first converted to binary data using one-hot encoding approach. Then autoencoder is used to further process the binary data. The details of the autoencoder are demonstrated in Section 3.1.

The second stage is the censored data processing. Censored data is common in historical maintenance dataset. Cox PHM is a statistical model which is used to process censored and uncensored data [33]. The data obtained from autoencoder is combined with the numeric data and yield a new dataset. The features in the new dataset are then used as covariates. The new dataset is used to build a Cox PHM which can reveal the relationship between survival time and reliability. The difference in reliability between corrective maintenance and preventive maintenance is case dependent, which needs to be determined in the actual case. With the difference in reliability, the difference in TBF between corrective maintenance and preventive maintenance can be estimated, and the censored data is compensated. The compensated labels of preventive maintenance instances are closer to their actual TBF, compared to their original label. The compensated censored data is used jointly with the uncensored data to train an LSTM model in the next stage. The details of Cox PHM are introduced in Section 3.2.

Finally, after the data is pre-processed, LSTM network, a deep learning model, which is specialized in processing the sequential data, is used to train a prediction model. LSTM network is used to predict the next TBF based on the previous failure information. The details of the LSTM network will be introduced in Section 3.3. The flow chart of the proposed method is shown in Figure 1.

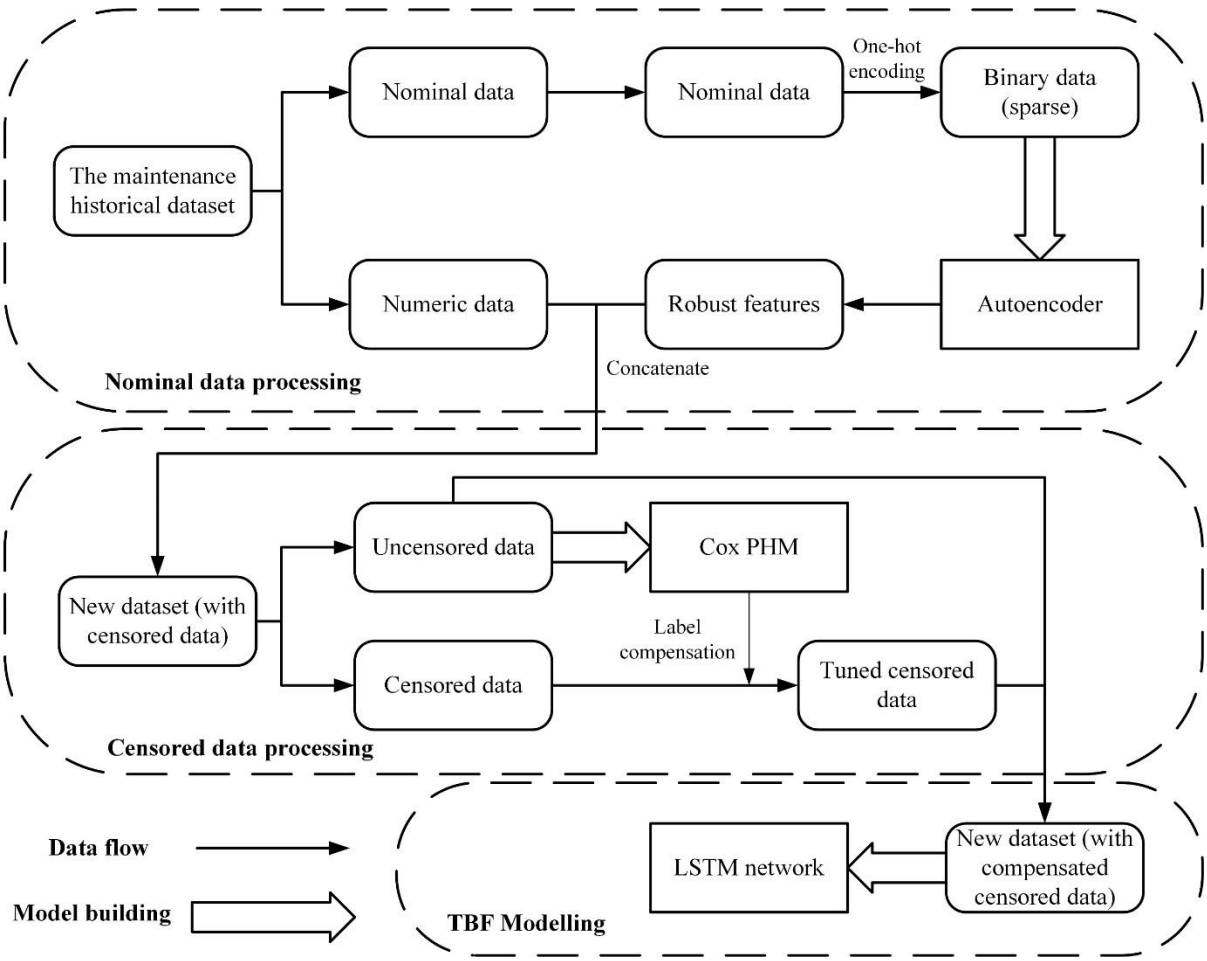

Figure 1. The flow chart of the proposed approach

3.1. Autoencoder

Autoencoder is an unsupervised learning technique, which has shown merits in feature extraction. It aims to learn the most significant features from data. The learned features are expected to reconstruct the original input completely [49]. An autoencoder consists of two parts which are encoder and decoder. An autoencoder can be described as a multi-layer neural network. The input layer and the first half of the hidden layers constitute the encoder, and the second half of the hidden layers and the output layer constitute the decoder. The number of nodes in each hidden layer is less than the number of nodes in the input layer and the output layer.

The input vector of the autoencoder is denoted as $\boldsymbol{x}$. The features learned by the encoder, also known as code, is denoted as $\boldsymbol{z}$.

The relation between $\boldsymbol{x}$ and $\boldsymbol{z}$ can be denoted as:

$$\boldsymbol{z} = \sigma(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}) \tag{1}$$

where $\boldsymbol{W}$ is the weight matrix between the input layer and the hidden layer, $\boldsymbol{b}$ is the bias, and the $\sigma(\cdot)$ is the activation function.

The features $\boldsymbol{z}$ learned from the hidden layer is then used to construct a vector $\boldsymbol{x}'$ which is expected the same as vector $\boldsymbol{x}$. The relationship between $\boldsymbol{x}'$ and $\boldsymbol{z}$ can be represented as:

$$\boldsymbol{x}' = \sigma[\boldsymbol{W}'\boldsymbol{z} + \boldsymbol{b}'] \tag{2}$$

where $\boldsymbol{W}'$ is the weight matrix between the input layer and the hidden layer, $\boldsymbol{b}'$ is the bias, and the $\sigma()$ is the activation function.

As autoencoder is a type of neural network, the parameters of $(\boldsymbol{W}, \boldsymbol{b})$ and $(\boldsymbol{W}', \boldsymbol{b}')$ can be trained via back-propagation algorithm. However, in the actual modelling, the vector $\boldsymbol{z}$ cannot be completely the same as vector $\boldsymbol{x}$. The difference between vector $\boldsymbol{x}$ and vector $\boldsymbol{x}'$ can be measured by a loss function. The adoption of the loss function is data-dependent. The structure of a three-layer autoencoder is shown in Figure 2.

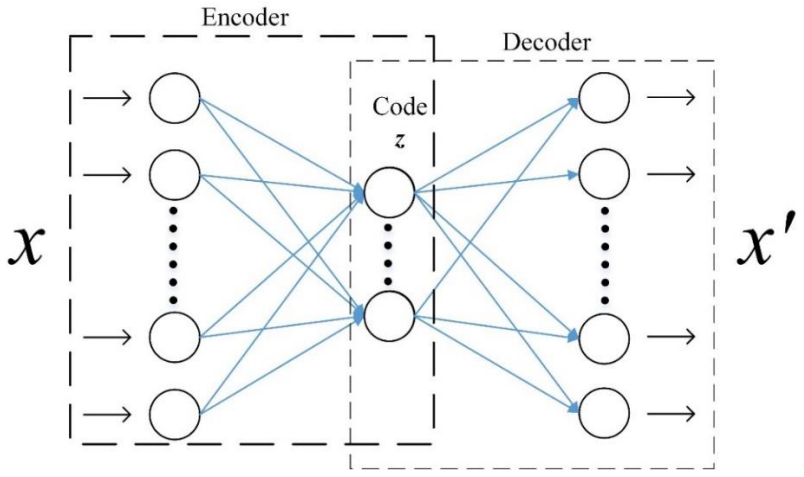

Figure 2. The structure of autoencoder

3.2. Cox PHM

Cox PHM is a statistical model which aims to analyse the relationship between time-independent covariates and hazard function [33]. The baseline hazard function is denoted as $\boldsymbol{h_0(t)}$. The covariate is denoted as $\boldsymbol{\beta_p}$ and the input vector is denoted as $\boldsymbol{X_p}$**.** The Cox PHM is denoted as:

$$\boldsymbol{h(t, X) = h_0(t)\, exp(\beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p)} \quad (3)$$

With the different adoption of $\boldsymbol{h_0(t)}$, Cox PHM can be parametric or non-parametric. A widely used function for the $\boldsymbol{h_0(t)}$ is the maximum likelihood estimator proposed by Breslow [50].

Figure. 3 shows a reliability curve generated by Cox PHM. The actual TBF of the censored data needs to be estimated. The difference in reliability $\boldsymbol{\Delta R}$ between corrective maintenance and preventive maintenance is assumed the same for all the equipment possessed by the same company. Hence, once $\boldsymbol{\Delta R}$ can be determined, it can be used to determine the difference of TBF $\boldsymbol{\Delta T}$ between corrective maintenance and preventive maintenance. $\boldsymbol{\Delta R}$ strongly depends on the maintenance strategy strategy of the fleet management company which is hard to be determined. The method used to determine $\boldsymbol{\Delta R}$ needs to be determined in the actual case. $\boldsymbol{\Delta R}$ is used to yield $\boldsymbol{\Delta T}$ for the censored data. Then $\boldsymbol{\Delta T}$ is used to compensate the TBF of the censored data. The censored data with compensated TBF is then used to modelling jointly with the uncensored data. The performance is noted and compared to determine the suitable value of $\boldsymbol{\Delta R}$.

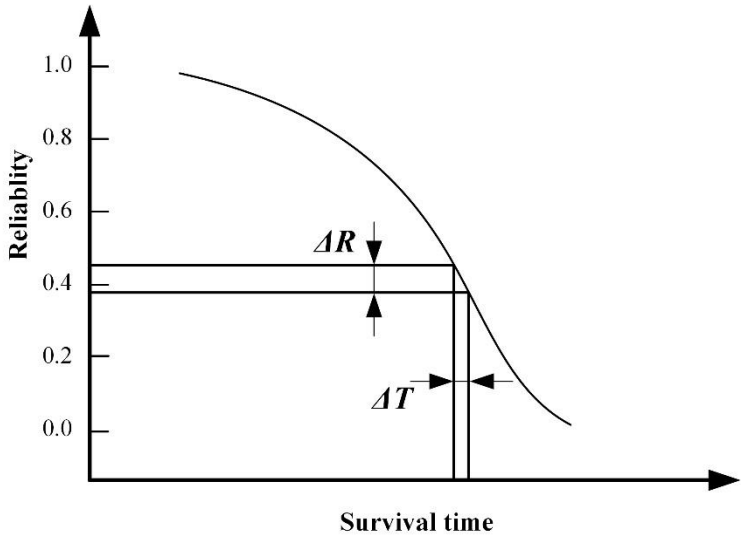

Figure. 3. The reliability curve generated by Cox PHM

3.3. Long-short-term Memory Network

LSTM network is a type of deep learning model which is well known for processing the sequential data. An LSTM layer has numerous cells. There are three gates, which are forgotten gate, input gate and output gate, used to control memory in each cell. The structure of an LSTM cell is shown in Figure 4.

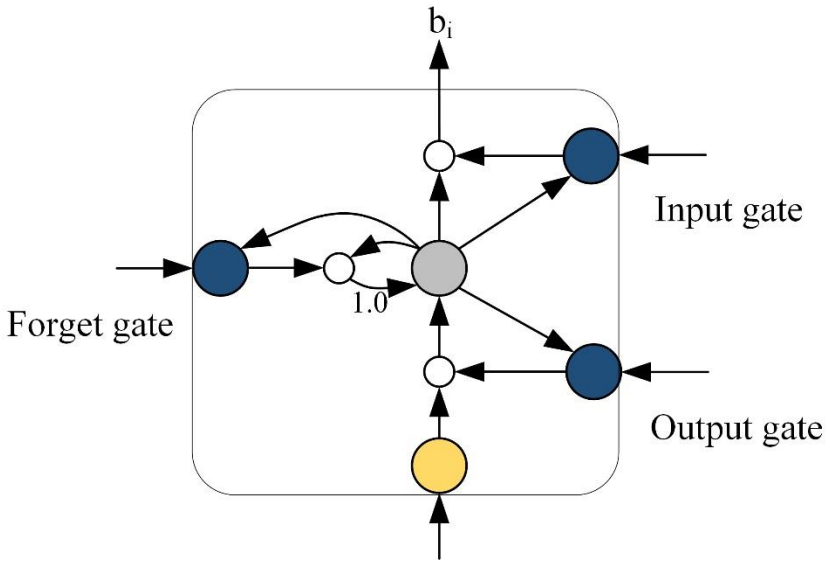

Figure 4. The structure of an LSTM cell

The $\boldsymbol{a_i}$ and $\boldsymbol{b_i}$ is the input and output of the LSTM memory cell. When $\boldsymbol{a_i}$ is transmitted into the LSTM AAAmemory cell, it is first processed by an activation function. The output of the activation function is then multiplied by $\boldsymbol{b_\iota}$. Secondly, the activation function output of the previous time step is multiplied by $\boldsymbol{b_\Phi}$. The product is added to the memory. Finally, the output of the memory is multiplied by $\boldsymbol{b_\omega}$ and then transmitted to another activation function to produce $\boldsymbol{b_i}$. The factors $\boldsymbol{b_\iota, b_\Phi, b_\omega}$ are represented by three white circles in Figure 4. These three factors are determined by the input gate, forget gate, and output gate respectively.

The TBF tends to be shorter with the increase of maintenance frequency. Our previous study has demonstrated that the TBF after $\boldsymbol{n^{th}}$ maintenance can be predicted using the automobile information collected in $\boldsymbol{n^{th}}$ maintenance [48]. However, the TBF of an automobile after $\boldsymbol{n^{th}}$ maintenance is not only relevant to the automobile information collected in $\boldsymbol{n^{th}}$ maintenance, while the previous maintenance information (before $\boldsymbol{n^{th}}$) also can be relevant to the TBF. Therefore, the information of the previous TBF and the previous maintenance information can be used for TBF modelling. Figure 5 illustrates an LSTM network for TBF modelling. The TBF after $\boldsymbol{n^{th}}$ maintenance is denoted as $\boldsymbol{y_{n+1}}$, the maintenance information collected in $\boldsymbol{nSAA^{th}}$ maintenance is denoted as $\boldsymbol{x_n}$, the previous maintenance information $\boldsymbol{X = [x_1, x_2, x_3, \dots, x_n]}$ is expected to be used for TBF modelling. An LSTM network is denoted as $f()$, the TBF modelling using LSTM network can be expressed as follow:

$$\boldsymbol{y_{n+1}} = f(\boldsymbol{x_1, x_2, x_3, \dots, x_n}) \tag{4}$$

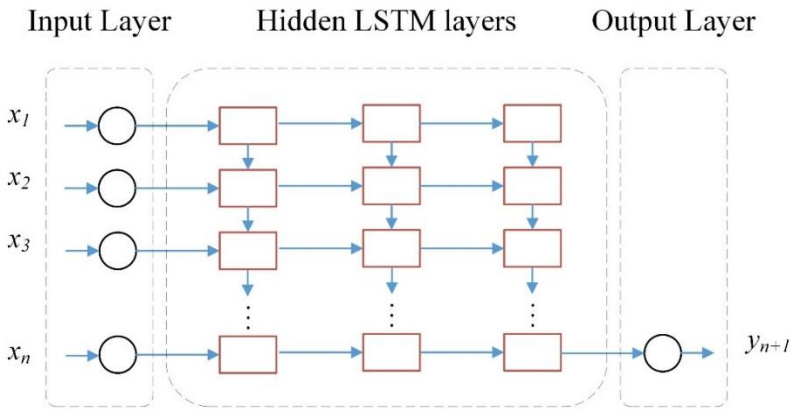

Figure 5. The diagram of an LSTM network for TBF modelling

In an LSTM network model, there are various parameters need to be determined, including the size of the network, optimiser, loss function, and learning rate, etc. Different parameters setting might result in different performance. The parameter setting is case dependent, which needs to be well selected. The performance of the proposed approach is evaluated in the following sections.

4. Experimental Setup

The historical maintenance dataset used in this case was provided by our industrial collaborator, which is a sizable fleet service company in the UK. The company has a keen interest in the TBF prediction of automobiles. An accurate prediction of TBF can offer insights for the fleet maintenance and further help the inventory management of replacement parts.

Firstly, it is worthwhile to provide general information on the company background. This company processes a large number of automobiles including various sizes of vans, personal cars, 4 by 4 vehicles. There are two types of maintenance management in the company. The first maintenance management type is run-to-failure (corrective maintenance), which is automobile is sent back to the workshop for maintenance when it actually breaks down. Workshop records the actual date of automobile failure, and therefore the actual TBF can be calculated. Another maintenance management type is preventive maintenance, which soon-to-failure is determined in the scheduled check. If an automobile is deemed will be failed in the near future during the scheduled check, the preventive maintenance is then carried out. The automobiles which experienced preventive maintenance is also recorded. However, the maintenance date in the dataset is earlier than the automobile's actual failure date which will result in the calculated TBF is shorter than the actual TBF. Thus, the TBF of the preventive maintenance record is right-censored.

Secondly, we focused on the procedure of data processing. With the domain knowledge, feature selection and data pre-processing can be implemented efficiently and accurately. Feature selection and

data pre-processing are introduced in Section 4.2. Thirdly, the metrics used to reveal performance and the validation method need to be considered.

Finally, in order to get comprehensive results from modelling, *k*-fold cross-validation was adopted. In this study, the value of *k* was set at 10.

4.1. Data

The data contains the maintenance record of the automobile engine. Each instance in the dataset represents one instance of a maintenance record. The data collection period had lasted for nearly nine years, from 2009 to 2017. There are over 12 thousand instances in the dataset. The quantity of censored data is 2,352, which takes 19.2% in the dataset. It can be seen that 40.9% of the automobile engines were failed within 500 days. The average of the TBF of all the instances in the dataset is 850.64 days.

The features relevant to automobile engine's lifecycle have been extracted from the dataset, which was used to build the TBF prediction model. The features are shown in Table 1. Among all the features, three of them are nominal and the rest are numeric. Due to the limitation of the number of features, all the features were selected for modelling. The numeric data can be directly used for modelling except for *isSch* because this feature cannot be determined before failure occurs. It is used to distinguish the right-censored data and the uncensored data in this case. Meanwhile, the nominal features, which are Model, Garage, and Area, are deemed highly relevant to the TBF according to the domain knowledge. One-hot encoding and autoencoder were used to further process the nominal data. The features mentioned above were selected to establish a TBF prediction model by adopting different machine learning algorithms. Figure. 6 shows the TBF prediction model with numeric and nominal features.

Table 1. The original feature relevant to TBF.

Numeric Feature	Note	Nominal Feature	Note
nRepair	The times of engine experienced maintenance	*Model*	The model of automobile
PAge	The age of automobile engine	*Garage*	The garage of automobile
VAge	The age of the automobile	*Area*	The area of automobile
CumM	The cumulative miles when a failure occurs		
Model_Year	The year of the first production		
Seq	A time index for automobile		
Regions	Four binary attributes		
isSch	A label whether a failure is found in schedule check		

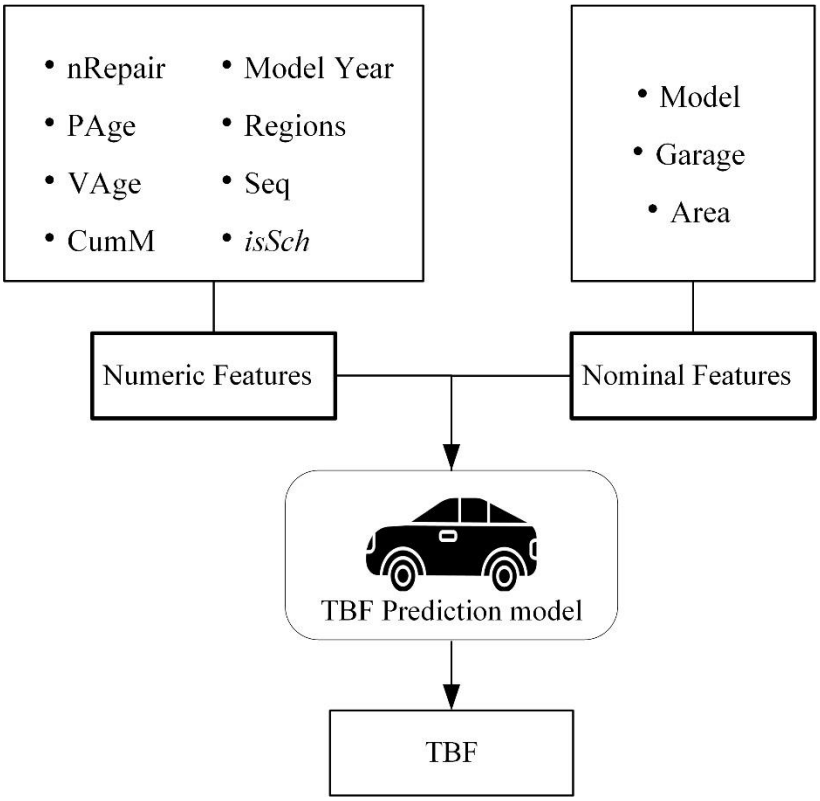

Figure. 6. TBF prediction model with features

The data was collected from the real world, which contains some impurity and noise in the dataset. The impurity and noise in the dataset were caused by the meter failure and meter misreading. The dataset with noise and impurity may damage the algorithm performance. Hence, the data pre-processing needs to be carried out. Firstly, abnormal and missing values were deleted. For example, there are approximately 30 records where the *CumM* of the automobile is over 4,000,000 miles, significantly

exceeding the mean *CumM* of 129,219 miles. These abnormal values were considered as the impurity which may be caused by meters failure or reading mistakes. Secondly, most of the missing values situated in the nominal features, which are hard to estimated and replaced. Hence, the instances which contain the abnormal and missing values in the dataset were removed. Thirdly, *Model Year* is a feature of point-in-time, which is meaningless to be analysed by machine learning algorithms. Hence, the difference between automobile registered year and its model year was used instead because it can represent the age of this automobile model. Finally, the data were normalised in order to improve data integrity and lower data redundancy. Since the nominal data had not converted in this stage, the normalisation was carried out later.

4.2. Model Setup

In the modelling stage, five machine learning algorithms, which are LSTM network, DCNN [24], recurrent neural network (RNN), artificial neural network (ANN) and support vector machine (SVM), are used for modelling. All the algorithms were used to train a TBF prediction model separately based on the pre-processed dataset. We adopted the default settings of the prevailing machine learning algorithm. The deep learning models were designed and established using Python Keras package [51]. The aim of modelling is to build a prediction model based on historical maintenance data. When designing a deep learning model, there are several issues need to be considered.

Firstly, the structure of deep learning model needs to be determined. The factors such as the type of layer, the number of layers, and the number of nodes in each layer directly affect the performance of the deep learning. If a model is designed extremely deep and large, it is able to predict TBF accurately. However, the computational cost will be extremely large as well. Hence, it is vital to balance the model complexity and computational cost. After several trials, three LSTM layer, one fully connected layer and three dropout layers were adopted as hidden layers in our LSTM network model. The size of the hidden nodes in LSTM and fully connected layers were set at 1000. Drop out layer is used to prevent

overfitting by disconnecting a certain percentage of nodes in the training process. In this case, the percentage was set at 20%. The deep learning model designed in this study is an LSTM network model. Secondly, the components and parameters of the LSTM network model, such as optimiser and loss function need to determined according to the data type and the aim of the study. The optimizer is used to optimise the learning process of the LSTM network model. RMSprop is an optimizer which is suitable for LSTM network, and therefore it was adopted. The loss function is set to be the mean squared difference between the actual value and prediction value.

The parameters relevant to the training process are the number of lookbacks, learning rate, batch size, and epochs. The number of lookbacks is the number of states that the LSTM network is considered at the same time. In this case, it was set at two, which means the variables in time ***t-1*** and time ***t-2*** are used to predict the output of time ***t*** due to the failure times of approximately 85.3% instances are lower than 2. The learning rate is the stride of the training process. In order to enable the LSTM network to sophisticatedly learn the hidden patterns of data, the learning rate was set at 0.001. The batch size was set at 150, which means 150 instances were fed into the LSTM network each time. The number of epochs was set at 45, which means the back-propagation process was repeated for 45 times to tune the parameters of the LSTM network. Meanwhile, the configuration of the ANN model in this study was basically the same as the LSTM model, except all the layers of the ANN are fully connected layers.

In order to tackle the data sparsity issue of one-hot encoded data, autoencoder, another deep learning algorithm, was introduced in this study. Autoencoder is a neural network which can extract the most significant features from sparse data [49]. The autoencoder designed in this case was a three-layer neural network comprising of an input layer, hidden layer, and output layer. The number of nodes of input and out layers were set at 160 which equals the size of one-hot encoding features. The number of nodes in the hidden layer is equal to the expected dimensions, which will be determined in the actual case. $\ell 1$-norm is a term that can be used to improve the prediction quality and its interpretability of modelling based on sparse data. It can be embedded in autoencoder to enhance its capability [11]. Because the one-hot encoding data is sparse, $\ell 1$-norm was introduced in the autoencoder in this case.

The parameter setting of the autoencoder was different from the deep learning algorithms used for modelling. Firstly, due to the output of the autoencoder is binary, and therefore Adadelta, a prevailing optimiser was adopted. Secondly, binary-cross-entropy was chosen as the loss function. Thirdly, the learning rate was set at the default value in Adadelta, which is 1. The batch size was set at three and the number of epochs was set at 30.

In this study, three scenarios were introduced. In scenario 1, prevailing machine learning and LSTM network were used for modelling based on the sparse data in conjunction with numeric data in historical maintenance dataset. In scenario 2, autoencoder was introduced to convert the one-hot encoding data to low dimension and robust data. Then the prevailing deep learning algorithms were used for modelling based on the robust data in conjunction with numeric data in historical maintenance dataset. Meanwhile, the relation between the algorithm performance and the number of the robust data dimension was explored. In scenario 3, based on the techniques in scenario 2, Cox PHM was introduced to tackle data censoring. In order to explore the impact of data sparsity on the algorithm performance of Cox PHM, two control experiments were set in scenario 3. In the first experiment, data compensation was based on a Cox PHM which was trained by sparse data in conjunction with the numeric data in historical maintenance dataset. In the second experiment, data compensation was based on Cox PHM which was trained by robust data in conjunction with the numeric data in historical maintenance dataset. After the censored data was compensated, the compensated censored data and uncensored data was jointly used for modelling using different algorithms. Also, the relation between the difference in reliability and the algorithm performance was explored in this scenario.

4.3. Performance Evaluation

Different metrics are needed to evaluate the algorithm performance from different perspectives. In this study, two metrics called Model correlation coefficient (MCC) and root-mean-square-error (RMSE) were chosen to evaluate the performance of algorithms. Both metrics have been widely used to evaluate

the results of the regression. They can reveal the algorithm performance in different perspectives. MCC is used to measure the correlation between two variables, and can be expressed mathematically as:

$$\text{MCC} = \frac{S_{PA}}{\sqrt{S_P S_A}}, \tag{5}$$

where,

$$S_{PA} = \frac{\sum_i (p_i - \bar{p})(a_i - \bar{a})}{n-1};\ S_P = \frac{\sum_i (p_i - \bar{p})^2}{n-1};$$

$$S_A = \frac{\sum_i (a_i - \bar{a})^2}{n-1};$$

p_i is the predicted value and $\bar{p}$ is the average of the predicted value. a_i is the actual value and the $\bar{a}$ is the average actual value. n is the number of training data.

It is a scale-dependent metric which measures the difference between the prediction value and the actual value. RMSE is 0 if the prediction value equals to the actual values. The expression of RMSE is:

$$\text{RMSE} = \sqrt{\frac{\sum_i (p_i - a_i)^2}{n}} \tag{6}$$

5. Results and Discussion

5.1. Scenario 1: Prevailing Machine Learning Algorithms VS. LSTM network

In this scenario, four deep learning and one prevailing machine learning algorithms were used for modelling based on numeric and one-hot encoding data. One-hot encoding was used to convert the nominal data to binary data, which can be processed by machine learning algorithms. There are three nominal features in the dataset which can be converted to 160 different categories using one-hot encoding technique. The one-hot dataset was then concatenated with the other numeric data in historical maintenance dataset. One-hot encoding enables the nominal data to be converted to numeric form without any information sacrifice. However, With the one-hot encoding features, there was a large number of 0 in the dataset which leads to significant sparsity in the dataset. After 10-fold cross-validation, the mean and standard deviation (STD) of MCC and RMSE were compared to reveal the algorithm performance. All tests were conducted on an Intel i5-6500 3.20Ghz PC with Nvidia GeForce GTX 1060 graphics card. The training time for each algorithm was marked and used to reveal the computational cost.

The modelling results based on the one-hot encoding data are shown in Table 2, which indicate that the LSTM network achieved the highest MCC which is 0.8248 and the lowest RMSE which is 379.8 days. SVM shows the worst performance in this scenario, which RMSE is 432.4 days and MCC is 0.7738. The algorithm performance of LSTM network in terms of MCC and RMSE are better than other algorithms. Moreover, the STD of MCC and RMSE of DCNN are the lowest in this scenario. Although the algorithm performance of LSTM network in terms of MCC and RMSE are better than the benchmarking algorithms, it also requires the longest training time.

Table 2. The results of machine learning modelling based on one-hot encoding data.

	LSTM Network	RNN	ANN	DCNN	SVM
MCC_Mean	**0.8248**	0.8221	0.8240	0.8240	0.7738
MCC_STD	0.0122	0.0136	0.0101	**0.0097**	0.0173
RMSE_ Mean (days)	**379.8**	382.1	387.2	387.2	432.4
RMSE_ STD (days)	14.78	13.27	16.62	**12.59**	15.91
Modelling time (s)	259.2	107.5	34.65	43.15	**7.263**

5.2. Scenario 2: Modelling based on Features Converted by Autoencoder

After the nominal data was converted using autoencoder, the one-hot encoding data was then combined with numeric data in historical maintenance dataset to generate a new dataset. The relation between the number of converted features and the algorithm performance in terms of MCC and RMSE are shown in Figure 7 and Figure 8.

It can be seen that the algorithm performance of all deep learning algorithms in terms of MCC and RMSE fluctuated in the beginning, and then become worse along with the larger number of converted features. In contrast, the algorithm performance in terms of MCC and RMSE of SVM is relatively stable. The algorithm performance in terms of MCC and RMSE of all algorithms reached their lowest points when the number of converted features ranges from 10 to 20. With the consideration of computational cost in TBF modelling stage and algorithm performance, the number of converted features is set at 16 in the following tests.

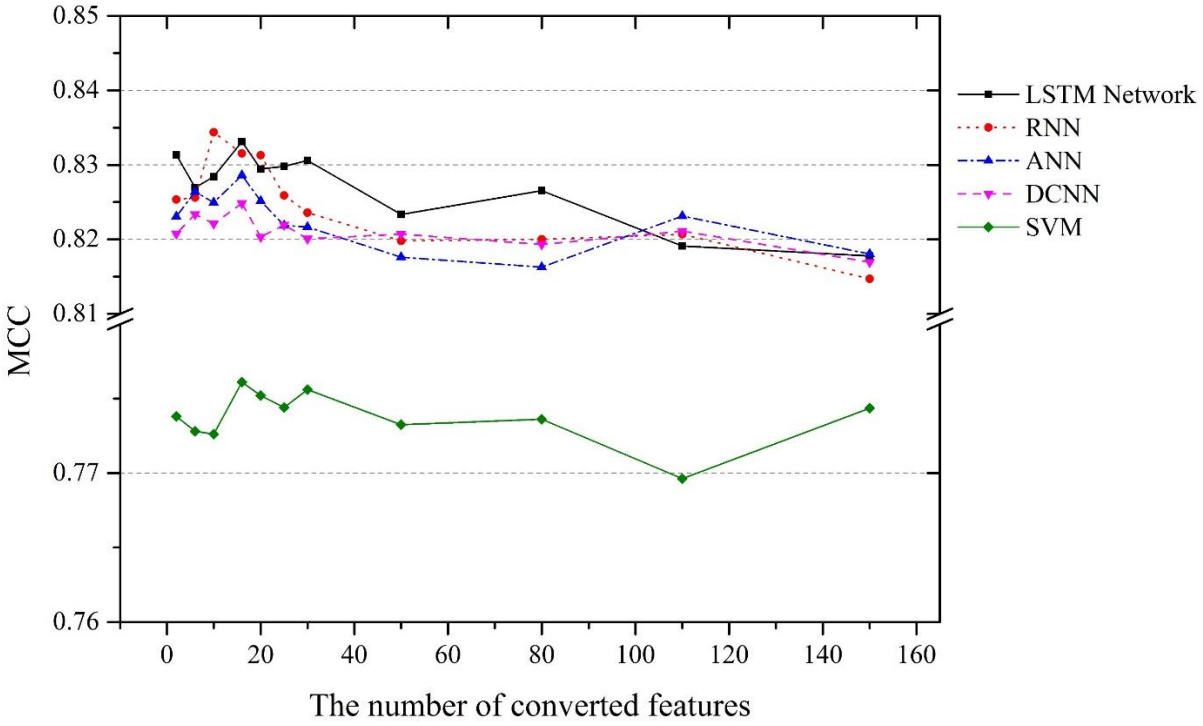

Figure 7. The relation between the number of converted features and the algorithm performance in terms of MCC

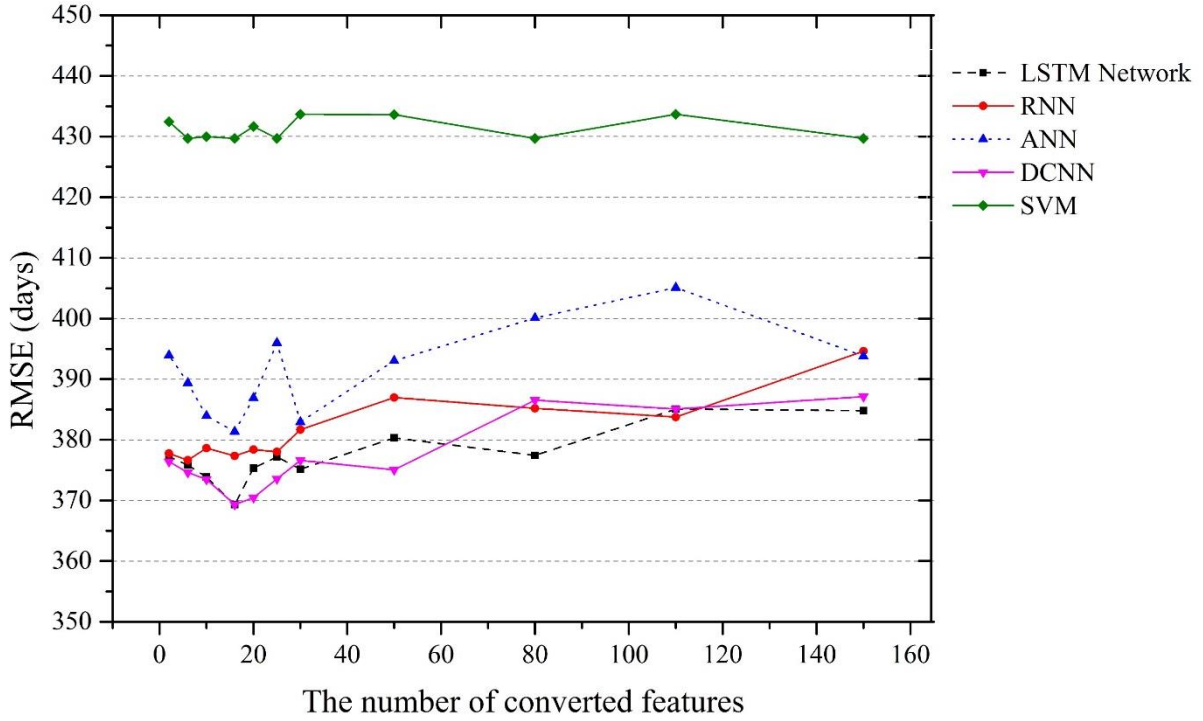

Figure 8. The relation between the number of converted features and the algorithm performance in terms of RMSE

After the number of converted features was determined. The converted data was concatenated with the numeric data in historical maintenance dataset and used for modelling. The results of the comparison between the modelling based on one-hot encoding and autoencoding are shown in Figure 9 and Figure 10. With the help of autoencoder, the algorithm performance in terms of RMSE of all the algorithms witnessed a decrease. LSTM network has still achieved the highest MCC which is 0.8348 and the lowest RMSE which is 369.3 days. When the autoencoder was introduced to convert the one-hot encoding data, the improvement of MCC and the decline of RMSE of the LSTM network are 0.91% and 1.84% respectively. The performance of the RNN in terms of RMSE increased by 1.13% and 2.10%, respectively. In contrast, the performance of SVM in terms of MCC and RMSE are merely increased with the help of autoencoder, while the STD of both MCC and RMSE are decreased dramatically. Finally, the STD of all the algorithms declined with the help of autoencoder. Hence, the results demonstrated that the robust features generated by autoencoder are helpful to improve performance in terms of MCC, RMSE and the stability to all the algorithms used in this study.

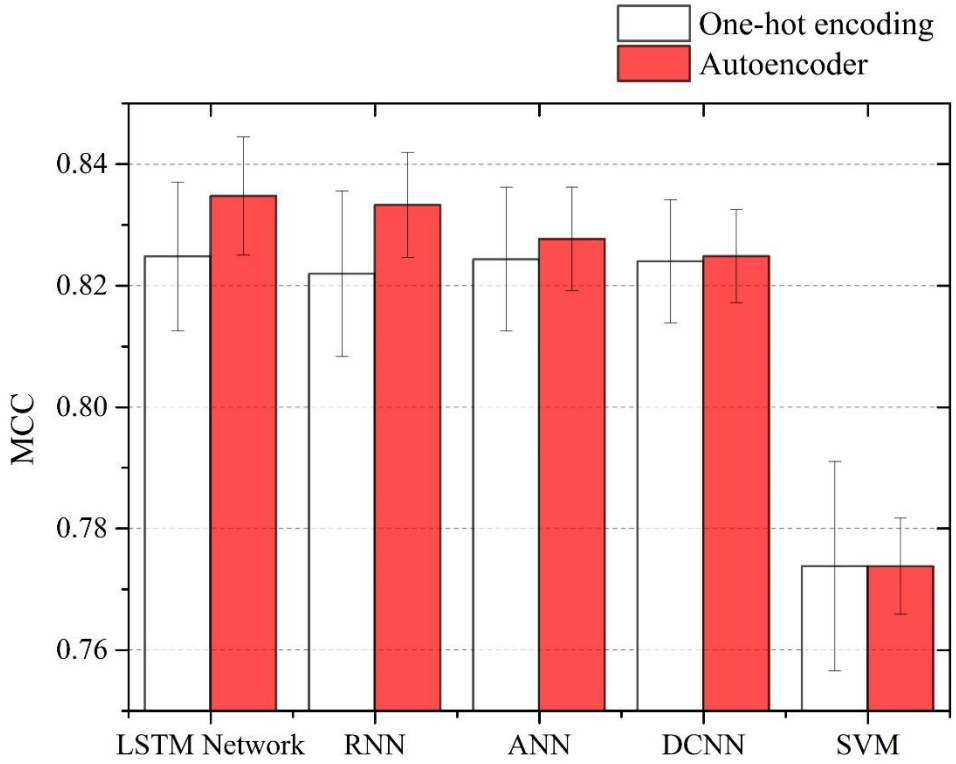

Figure 9. The algorithm performance comparison between one-hot encoding-based modelling and autoencoder based modelling the in terms of MCC

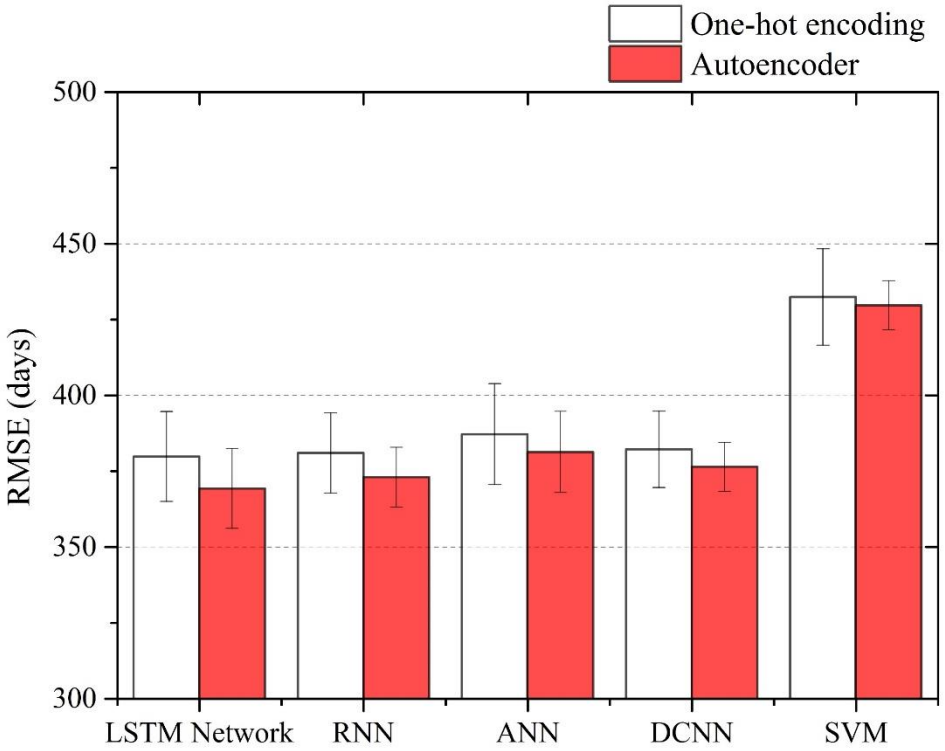

Figure 10. The algorithm performance comparison between one-hot encoding based modelling and autoencoder based modelling the in terms of RMSE

5.3.*Scenario 3: Modelling Based on Cox Proportional Hazard Deep Learning*

Data censoring leads to the inaccurate label of data. If the censored data is directly used for modelling, it may jeopardise the algorithm performance. If an appropriate compensation for the censored data can be estimated, the algorithm performance could be promoted. There are 2,352 censored instances in this study, which takes 19.2% in the dataset. In order to estimate the actual TBF for the censored data, Cox PHM was introduced into this study. Uncensored data was used to build a Cox PHM.

Appropriate compensation based on the difference in reliability can be beneficial to performance. With Cox PHM, the relationship between reliability and TBF can be estimated. However, when the difference in reliability is too large, the algorithm performance tends to be damaged due to the tuned TBF is inaccurate. The difference in reliability was used to generate compensation for the censored data according to Cox PHM of each censored instance. When a difference in reliability is set, then the

compensation of each censored instance can be determined. The censored data with compensation was then used for modelling jointly with the uncensored data. In this case, the ideal difference in reliability needs to be determined. The difference in reliability was first set in the range of 0% to 5%. If the algorithm performance can be promoted when the difference in reliability increases from this range, it would then be expended to find the optimal point. Two control experiments were conducted in this scenario to reveal the impact of data sparsity on the algorithm performance of Cox PHM. The results of the first experiment are shown in Figure 11 and Figure 12.

A Cox PHM was trained by sparse data in conjunction with the numeric data in historical maintenance dataset, which was used for label compensation. It can be seen from Figure 11 that the fluctuation of all the deep learning algorithms is considerable. The algorithm performance of SVM in terms of MCC reached its peak when the difference in reliability is 2%, following by a monotonous fall. Also obvious is that, with suitable label compensation, the algorithm performance of different algorithms in terms of MCC were promoted. LSTM network achieved the highest MCC is this experiment, which is 0.8383. With the help of label compensation, the maximum MCC improvement of LSTM network, RNN, ANN, DCNN and SVM are 0.51%,0.55%, 0.31%, 0.19% and 0.37%, respectively. From Figure 12, it is evident that the algorithm performance in terms of RMSE for all the algorithms fell to a low point before a monotonous increase. LSTM network achieved the lowest RMSE in this figure which is 364.5 days. RNN witnessed the largest algorithm performance improvement in terms of RMSE, which is 1.80%.

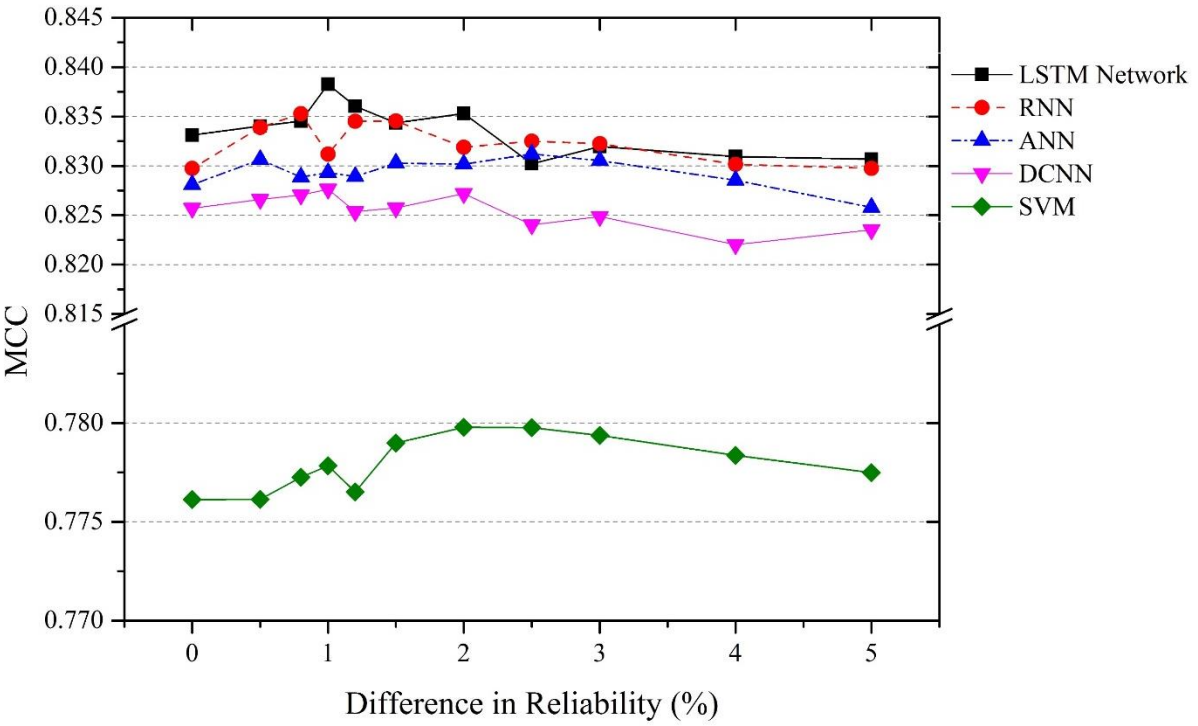

Figure 11. The MCC of modelling based on Cox PHM trained by sparse data in conjunction with numeric data in historical maintenance dataset

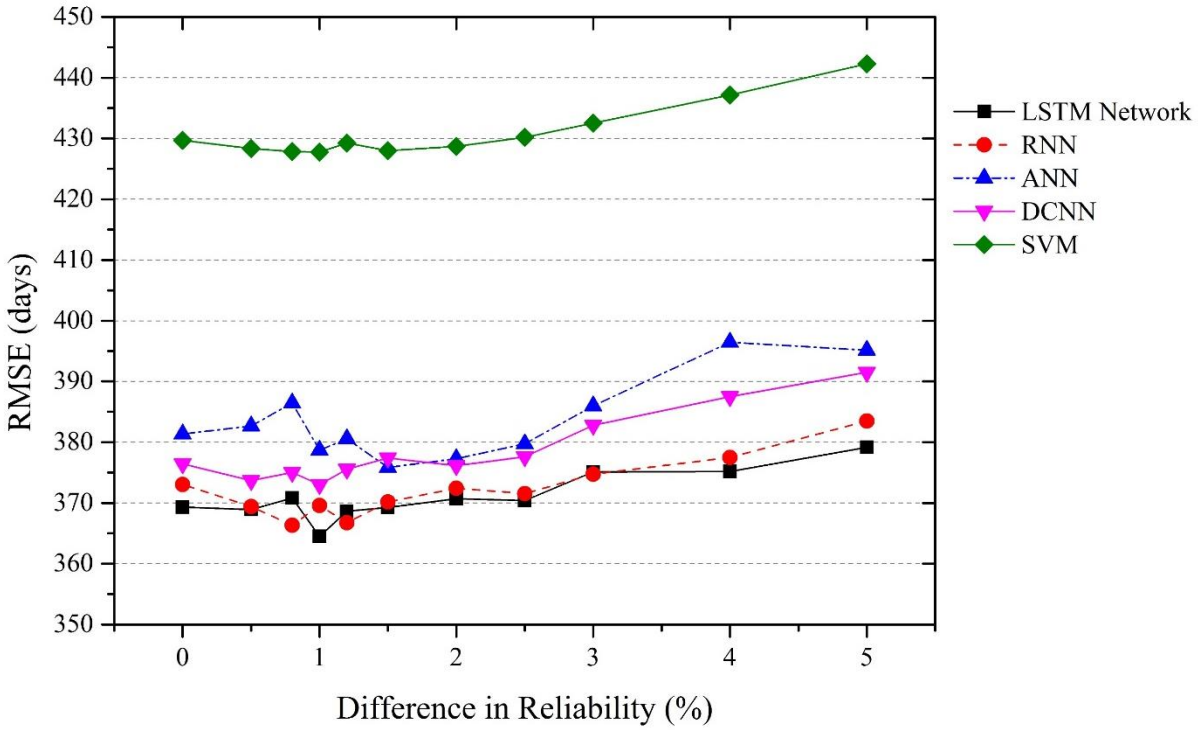

Figure 12. The MCC of modelling based on Cox PHM trained by sparse data in conjunction with numeric data in historical maintenance dataset

With the robust features getting from scenario 2, a reliable Cox PHM trained by the robust data in conjunction with the numeric data in historical maintenance dataset was established. With the consideration of the impact of data sparsity on the algorithm performance of Cox PHM, the robust data in conjunction with the numeric data was used to train a reliable Cox PHM for label compensation. The modelling results are shown in Figure 13 and Figure 14. It can be seen from Figure 13 that the MCC of all algorithms are leveraged when the difference in reliability is chosen appropriately. The MCC of SVM has been increased to the highest point when the difference in reliability grows from 0% to 1.2%. In contrast, the performance of all deep learning algorithms in terms of MCC fluctuated slightly in all stages. The MCC of all deep learning algorithms firstly increased and then reached their peaks when the difference in reliability ranged from 0.5% and 2%, followed by a period of continuously decrease. LSTM network achieved the highest MCC in this scenario, which is 0.8395. The MCC of the LSTM network is higher than that of other algorithms in all the stage. It is also clear in Figure 14 that all the algorithms experienced a similar trend when the difference in reliability grows from 0% to 5%. The RMSE of LSTM network, RNN, ANN and CNN reach their lowest point when the difference in reliability is 0.8% or 1%. The lowest RMSE in this scenario is 359.1 days, which is achieved by LSTM network. Meanwhile, the performance of SVM in terms of RMSE become better when the difference in reliability ranges from 0.3% to 1.2%. The maximum RMSE decline in terms of RMSE of LSTM network, network, RNN, ANN, DCNN and SVM is 2.75%, 2.56 %, 1.78%, 1.61% and 0.96% respectively. Hence, in terms of MCC and RMSE, the performance of all the algorithms was promoted with the help of the Cox PHM trained by in conjunction with the numeric data in historical maintenance dataset.

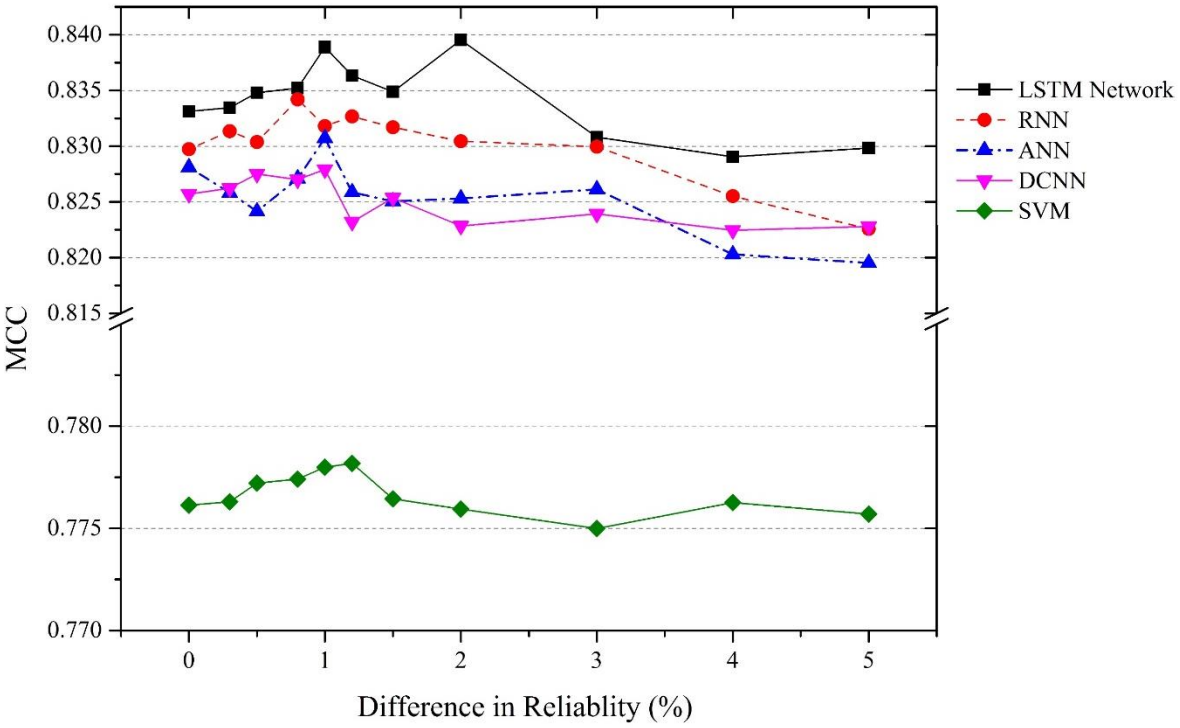

Figure 13. The MCC of modelling based on Cox PHM trained by robust data in conjunction with numeric data in historical maintenance dataset

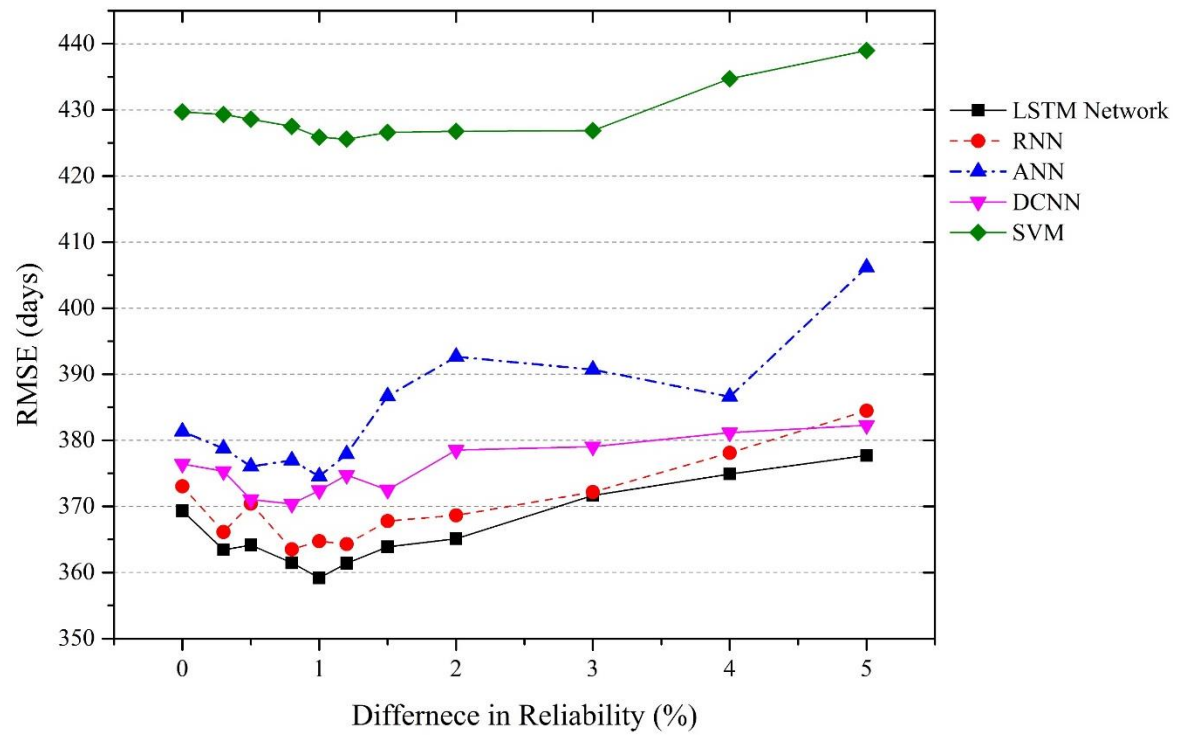

Figure 14. The RMSE of modelling based on Cox PHM trained by robust data in conjunction with numeric data in historical maintenance dataset

Figure 11-14 demonstrated that the algorithm performance in terms of MCC and RMSE of all the algorithms were benefited more from the Cox PHM trained by in conjunction with the numeric data, which indicates that the data sparsity jeopardises the algorithm performance of Cox PHM. Since LSTM achieved the best algorithm performance in terms of MCC and RMSE, a further algorithm comparison is shown in Figure 15. It can be seen that with the solutions of data sparsity and data censoring, the algorithm performance of LSTM network in terms of MCC was promoted by 1.8% and RMSE was reduced by 5.4%. Besides, the standard deviation of MCC and RMSE in 10-fold cross-validation was shrunk with the help of autoencoder and Cox PHM.

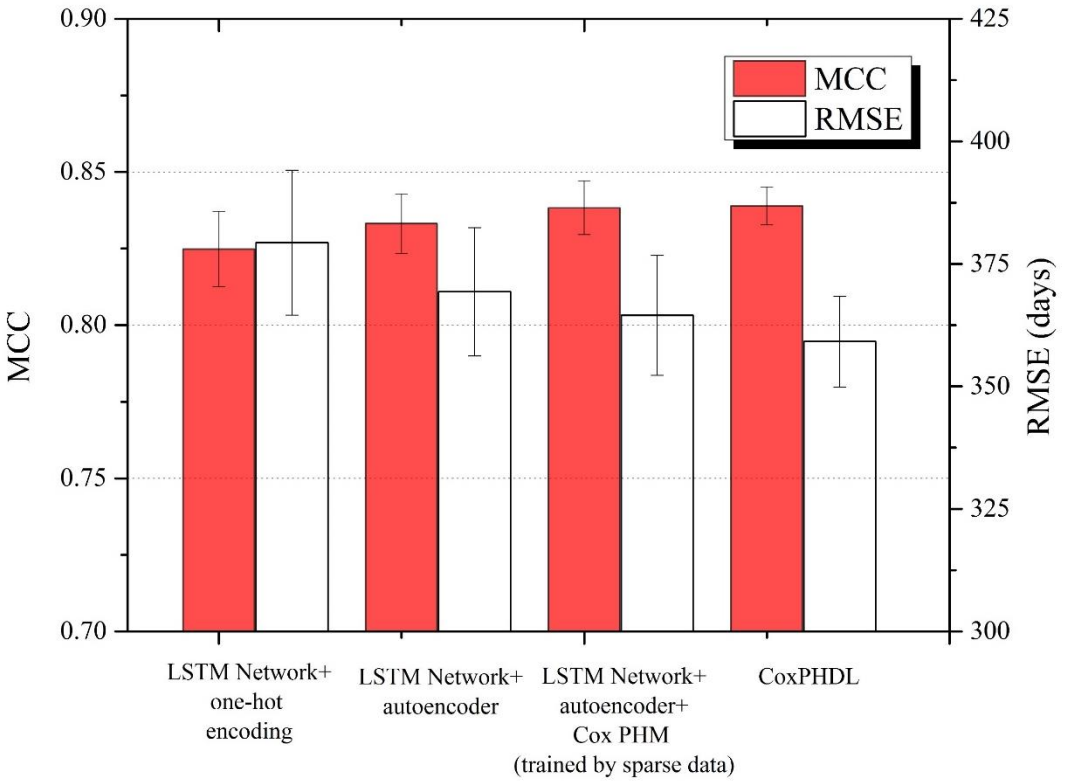

Figure 15. The algorithm performance of LSTM in different schemes in terms of MCC and RMSE

6. Discussion

PdM is of importance to the industry. The issues of maintenance planning, job scheduling and spare parts inventory management have long been concerned by various industries such as fleet management.

With the accurate prediction of TBF, better fleet management can be to achieve. Most researches in PdM have been conducted using sensor data. However, sensor data collection requires extra expenditure, which is unaffordable for some companies. Different from the existing researches, this study focuses on PdM based on the historical maintenance data, which is relatively easy to obtain in the industry.

6.1. Results Discussion

The results of scenario 1 indicate that the algorithm performance is positive to the computational cost in this study. The LSTM networks achieved the highest MCC and lowest RMSE in scenario 1, while it took the longest training time in comparison with the prevailing machine learning algorithms. What is noticeable in the results of scenario 2 is that the robust data representation converted by autoencoder is useful to improve the algorithm performance of all the algorithms used in this study, which indicates that data sparsity damage algorithm performance in terms of the MCC and RMSE of all the algorithms used in this study. The algorithm performance in terms of MCC and RMSE worsen along with the larger number of converted features. With this knowledge, the autoencoder can be better deployed in the future study.

The results of scenario 3 firstly indicate that the data sparsity also has a negative impact on the algorithm performance of Cox PHM. With the robust features obtained from autoencoder, the effect of label compensation using Cox PHM was enhanced. After the conversion of the categorical variables using autoencoder and the estimation of the label of censored data using CoxPHM, the algorithm performance of the standard LSTM network in terms of MCC and RMSE are generally better than all other algorithms, which indicates it is more sensitive to the data sparsity and data censoring issue than other algorithms used in this study. The proposed technique can improve the algorithm performance in terms of MCC and RMSE of LSTM network, which is 1.8% and 5.4% res. The relation between the difference in reliability and algorithm performance was revealed in this study. The ideal difference in reliability in this study ranges from 0.8% to 1.5%, which can offer some insights into the application of CoxPHDL.

6.2.Future Works

The algorithm performance in terms of MCC and RMSE can be continuously improved and decreased in our future research. In the historical maintenance dataset used in this study, we have noticed that some automobiles with a similar condition have different TBFs. In other words, there are some data which features are similar or even the same, while their labels are different. The main reason is that the features relevant to the automobile lifecycle were not sufficiently collected. The historical maintenance data may not contain sufficient patterns relevant to automobile engine degradation which enable the algorithm to yield a precise prediction model. It is well known that the TBF of the automobile depends on various factors including geographical environment, driving behaviour, and product design, etc. If these data can be collected and introduced into our study, the algorithm performance is likely to be further promoted. Moreover, the research on these data will also be helpful for the fleet management company to have a better understanding of how these factors impact the automobile lifecycle.

The Cox PHM in this study is under the assumption that the difference in reliability between corrective maintenance and preventive maintenance is the same for all automobiles owned by the same company. However, the difference in reliability of automobiles might slightly vary from each other. It can be caused by the following reasons. On the one hand, the data used in this case was collected from different garages of the fleet management company under investigation. The maintenance rules and standards could be slightly different in different garages. On the other hand, the preventive maintenance strategy deployed in the company's garages also depends on the engineer's judgment and experience which could lead to the variation of difference in reliability. If a better estimation of the difference in the reliability of each automobile can be achieved, it is possible to further lift the algorithm performance of LSTM network. Hence, a better approach used to estimate the difference in reliability needs to be investigated further.

7. Conclusions

PdM is important to various industries such as automobile, aircraft and railway. The prediction of TBF can bring tangible benefits to the industry so as to achieve better maintenance planning, job scheduling and spare parts inventory management. In this paper, the focus is on the modelling and prediction of TBF based on historical maintenance data. Relevant works including the application of machine learning techniques in PdM, the application of statistical techniques in PdM, and the techniques used to address data sparsity and censoring issues PdM were reviewed. Based on autoencoder, Cox PHM, and LSTM network, we have proposed a new approach call Cox proportional hazard deep learning (CoxPHDL) to predict TBF based on historical maintenance data. In this approach, autoencoder is used to convert the nominal data for Cox PHM and LSTM network. Cox PHM is used to estimate the label of censored data. After the data is pre-processed, LSTM network algorithm is used to build a TBF prediction model. An experimental study was carried out based on real-world automobile historical maintenance data. There are two key findings in our study. Firstly, LSTM network shows merits in TBF modelling in comparison with several prevailing machine learning algorithms, but it leads to a higher computation cost. Secondly, data sparsity shows a negative impact on the algorithm performance of Cox PHM and LSTM network. With the consideration of this issue, autoencoder was deployed to address this issue and promote the algorithm performance of LSTM network and most of the machine learning algorithms. Then, with the help of Cox PHM, the algorithm performance of LSTM network and most of the machine learning algorithms can be further leveraged. In the actual fleet management scenario, this is deemed very useful to improve the job scheduling, automobile maintenance planning and the inventory management of spare parts.

Acknowledgement

This research was partially supported by Engineering and Physical Sciences Research Council (EPSRC) UK under the project “Resilient Remanufacturing Networks: Forecasting, Informatics and Holons” (EP/P008925/1).

References

[1] R. Prytz, S. Nowaczyk, T. Rögnvaldsson, S. Byttner, Predicting the need for vehicle compressor repairs using maintenance records and logged vehicle data, Engineering applications of artificial intelligence, 41 (2015) 139-150.

[2] O.O. Aremu, D. Hyland-Wood, P.R. McAree, A Relative Entropy Weibull-SAX framework for health indices construction and health stage division in degradation modeling of multivariate time series asset data, Advanced Engineering Informatics, 40 (2019) 121-134.

[3] P. Baruah, R.B. Chinnam, HMMs for diagnostics and prognostics in machining processes, International Journal of Production Research, 43 (2005) 1275-1293.

[4] G.A. Susto, A. Schirru, S. Pampuri, S. McLoone, A. Beghi, Machine learning for predictive maintenance: A multiple classifier approach, IEEE Transactions on Industrial Informatics, 11 (2015) 812-820.

[5] Z. Allah Bukhsh, A. Saeed, I. Stipanovic, A.G. Doree, Predictive maintenance using tree-based classification techniques: A case of railway switches, Transportation Research Part C: Emerging Technologies, 101 (2019) 35-54.

[6] T. Edwards, A. Bayoumi, M.L. Eisner, Internet of Things–A Complete Solution for Aviation’s Predictive Maintenance, Advanced Technologies for Sustainable Systems, Springer2017, pp. 167-177.

[7] F. Civerchia, S. Bocchino, C. Salvadori, E. Rossi, L. Maggiani, M. Petracca, Industrial Internet of Things monitoring solution for advanced predictive maintenance applications, Journal of Industrial Information Integration, 7 (2017) 4-12.

[8] D.L. Nuñez, M. Borsato, OntoProg: An ontology-based model for implementing Prognostics Health Management in mechanical machines, Advanced Engineering Informatics, 38 (2018) 746-759.

[9] M. Cassel, F. Lima, Evaluating one-hot encoding finite state machines for SEU reliability in SRAM-based FPGAs, On-Line Testing Symposium, 2006. IOLTS 2006. 12th IEEE International, IEEE, 2006, pp. 6.

[10] E. Hosseini-Asl, J.M. Zurada, O. Nasraoui, Deep learning of part-based representation of data using sparse autoencoders with nonnegativity constraints, IEEE transactions on neural networks and learning systems, 27 (2016) 2486-2498.

[11] K. Han, C. Li, X. Shi, Autoencoder Feature Selector, arXiv preprint arXiv:1710.08310, (2017).

[12] T. Dumas, A. Roumy, C. Guillemot, Shallow sparse autoencoders versus sparse coding algorithms for image compression, Multimedia & Expo Workshops (ICMEW), 2016 IEEE International Conference on, IEEE, 2016, pp. 1-6.

[13] J. Anderson, A. Senthilselvan, A two-step regression model for hazard functions, Applied Statistics, (1982) 44-51.

[14] R. Kay, Proportional hazard regression models and the analysis of censored survival data, Applied Statistics, (1977) 227-237.

[15] D. Kumar, B. Klefsjö, Proportional hazards model: a review, Reliability Engineering & System Safety, 44 (1994) 177-188.

[16] L. Liao, H.-i. Ahn, Combining deep learning and survival analysis for asset health management.

[17] J. Qin, Y. Liu, R. Grosvenor, Multi-source data analytics for AM energy consumption prediction, Advanced Engineering Informatics, 38 (2018) 840-850.

[18] X.-J. Wan, L. Liu, Z. Xu, Z. Xu, Q. Li, F. Xu, Fault diagnosis of rolling bearing based on optimized soft competitive learning Fuzzy ART and similarity evaluation technique, Advanced Engineering Informatics, 38 (2018) 91-100.

[19] G.H. Kruger, A.J. Shih, D.G. Hattingh, T.I. van Niekerk, Intelligent machine agent architecture for adaptive control optimization of manufacturing processes, Advanced Engineering Informatics, 25 (2011) 783-796.

[20] Z. Wei, T. Tao, D. ZhuoShu, E. Zio, A dynamic particle filter-support vector regression method for reliability prediction, Reliability Engineering & System Safety, 119 (2013) 109-116.

[21] P.G. Nieto, E. García-Gonzalo, F.S. Lasheras, 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, Reliability Engineering & System Safety, 138 (2015) 219-231.

[22] D. Lee, R. Pan, Predictive maintenance of complex system with multi-level reliability structure, International Journal of Production Research, 55 (2017) 4785-4801.

[23] G. Zhao, X. Liu, B. Zhang, G. Zhang, G. Niu, C. Hu, Bearing Health Condition Prediction Using Deep Belief Network, Annual Conference of the Prognostics and Health Management Society (2017).

[24] X. Li, Q. Ding, J.-Q. Sun, Remaining useful life estimation in prognostics using deep convolution neural networks, Reliability Engineering & System Safety, 172 (2018) 1-11.

[25] A. Malhi, R. Yan, R.X. Gao, Prognosis of defect propagation based on recurrent neural networks, IEEE Transactions on Instrumentation and Measurement, 60 (2011) 703-711.

[26] M. Yuan, Y. Wu, L. Lin, Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network, Aircraft Utility Systems (AUS), IEEE International Conference on, IEEE, 2016, pp. 135-140.

[27] J. Zhang, P. Wang, R. Yan, R.X. Gao, Deep Learning for Improved System Remaining Life Prediction, Procedia CIRP, 72 (2018) 1033-1038.

[28] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature, 521 (2015) 436-444.

[29] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation, 9 (1997) 1735-1780.

[30] X.-W. Chen, X. Lin, Big data deep learning: challenges and perspectives, IEEE access, 2 (2014) 514-525.

[31] M. Xie, C.D. Lai, Reliability analysis using an additive Weibull model with bathtub-shaped failure rate function, Reliability Engineering & System Safety, 52 (1996) 87-93.

[32] A. Mettas, Modeling and analysis for multiple stress-type accelerated life data, Reliability and Maintainability Symposium, 2000. Proceedings. Annual, IEEE, 2000, pp. 138-143.

[33] D.R. Cox, Regression models and life-tables, Breakthroughs in statistics, Springer1992, pp. 527-541.

[34] T.L. Landers, H.E. Soroudi, Robustness of a semi-parametric proportional intensity model, IEEE Transactions on Reliability, 40 (1991) 161-164.

[35] Y. Sun, L. Ma, J. Mathew, W. Wang, S. Zhang, Mechanical systems hazard estimation using condition monitoring, Mechanical systems and signal processing, 20 (2006) 1189-1201.

[36] A.L. Samuel, Some studies in machine learning using the game of checkers, IBM Journal of research and development, 3 (1959) 210-229.

[37] S. Wang, Y. Liu, C. Di Cairano-Gilfedder, S. Titmus, M.M. Naim, A.A. Syntetos, Reliability Analysis for Automobile Engines: Conditional Inference Trees, Procedia CIRP, 72 (2018) 1392-1397.

[38] K. Bastani, P.K. Rao, Z. Kong, An online sparse estimation-based classification approach for real-time monitoring in advanced manufacturing processes from heterogeneous sensor data, IIE Transactions, 48 (2016) 579-598.

[39] X. Chen, G. Cai, H. Cao, W. Xin, Condition assessment for automatic tool changer based on sparsity-enabled signal decomposition method, Mechatronics, 31 (2015) 50-59.

[40] H. Liu, C. Liu, Y. Huang, Adaptive feature extraction using sparse coding for machinery fault diagnosis, Mechanical Systems and Signal Processing, 25 (2011) 558-574.

[41] B. Vaisakh, R. Gopinath, C.S. Kumar, M. Ganesan, Condition monitoring of synchronous generators using sparse coding, International Journal of Applied Engineering Research, 10 (2015) 26689-26697.

[42] R. Sipos, D. Fradkin, F. Moerchen, Z. Wang, Log-based predictive maintenance, (2014) 1867-1876.

[43] X. Jiang, Y. Yuan, X. Liu, Multivariate probabilistic modelling for seepage risk assessment in tunnel segments, International Journal of Reliability and Safety, 8 (2014) 228-249.

[44] S. Pampuri, A. Schirru, C. De Luca, G. De Nicolao, Proportional hazard model with ℓ1 Penalization applied to Predictive Maintenance in semiconductor manufacturing, 2011 IEEE International Conference on Automation Science and Engineering, IEEE, 2011, pp. 250-255.

[45] N. Balakrishnan, D. Mitra, Some further issues concerning likelihood inference for left truncated and right censored lognormal data, Communications in Statistics-Simulation and Computation, 43 (2014) 400-416.

[46] Y. Hong, W.Q. Meeker, J.D. McCalley, Prediction of remaining life of power transformers based on left truncated and right censored lifetime data, The Annals of Applied Statistics, 3 (2009) 857-879.

[47] J.A. Christen, F. Ruggeri, E. Villa, Utility based maintenance analysis using a Random Sign censoring model, Reliability Engineering & System Safety, 96 (2011) 425-431.

[48] C. Chen, Y. Liu, X. Sun, S. Wang, C. Di Cairano-Gilfedder, S. Titmus, A.A. Syntetos, Reliability Analysis Using Deep Learning, ASME 2018 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, 2018, pp. 1-10.

[49] C. Hong, J. Yu, J. Wan, D. Tao, M. Wang, Multimodal deep autoencoder for human pose recovery, IEEE Transactions on Image Processing, 24 (2015) 5659-5670.

[50] N.E. Breslow, Analysis of survival data under the proportional hazards model, International Statistical Review/Revue Internationale de Statistique, (1975) 45-57.

[51] F. Chollet, Keras: Deep learning library for theano and tensorflow, URL: https://keras.io, (2015).