

Automobile Maintenance Modelling Using gcForest

Chong Chen, Ying Liu, *Member, IEEE*, Xianfang Sun, Carla Di Cairano-Gilfedder and Scott Titmus

Abstract— Automobile maintenance has gained increasing attention in recent years. If the failure time of an automobile can be predicted, it can bring tangible benefits to automobile fleet management. The Multi-Grained Cascade Forest (gcForest) is a tree-based deep learning algorithm, which was originally developed for image classification, but is potentially a helpful tool in automobile maintenance. This study aims to introduce the gcForest into automobile maintenance based on automobile historical maintenance data and geographical information system (GIS) data. The experimental results reveal that the gcForest shows merits in automobile time-between-failure (TBF) modelling, while it requires less computational cost.

I. INTRODUCTION

Maintenance is a key issue in the era of smart manufacturing. Recently, with the availability of a large amount of data relevant to equipment lifecycle, predictive maintenance (PdM) has gained increasing attention in the industry [1, 2]. The prediction of the failure time of equipment is the key part of PdM, which aims to maximize the service life of equipment without sacrificing reliability. If the maintenance can be scheduled before a critical failure occurs, it can lower the maintenance cost and downtime. An automobile fleet management company can be beneficial from PdM in terms of spare parts management, job scheduling and maintenance planning.

PdM can be classified as two types according to the data used for modelling: (1) statistical PdM and (2) condition-based PdM [3]. Recently, a lot of researchers have focused on condition-based PdM, which aims at modelling the degradation trend of equipment based on sensor data. In contrast, the studies of statistical PdM, which are based on maintenance data, are very limited. Furthermore, most studies of statistical PdM have focused on classification mission, while most studies of condition-based PdM have focused on regression mission. Owing to the fact that the collection of sensor data tends to be more expensive in comparison with historical maintenance data [4], it is worthwhile to study how to establish a model for equipment failure time prediction based on historical maintenance data.

Deep learning is a group of prevailing machine learning algorithms [5]. It has been widely studied in PdM [6-10]. Most deep learning algorithms such as convolutional neural network, recurrent neural network, and deep brief network were proposed based on neural network. These algorithms have been widely used in processing images, audio and text data [5]. The decision tree is another important type of machine learning algorithm. Zhou and Feng (2017) [11]

proposed a novel deep learning algorithm called deep forest (gcForest), which is a decision tree ensemble approach that requires fewer parameters and can generate decent algorithm performance in comparison with the deep neural networks.

Various factors including weather, terrain and traffic can impact the lifecycle of an automobile. These data can be captured via geographical information system (GIS) [12]. The prediction of the automobile failure date is important to fleet management. This study aims to establish an automobile TBF prediction model using gcForest algorithm based on automobile historical maintenance data and GIS data. Furthermore, how the different factors impact the automobile TBF is explored.

The rest of the paper is organized in the following sections: The existing researches of PdM and the studies of gcForest are reviewed in Section 2. Section 3 focuses on the overall methodology and the algorithm details of gcForest. Section 4 introduces an experimental study of TBF modelling using data collected from the real-world. Section 5 concludes the paper.

II. LITERATURE REVIEW

A. The Studies of Predictive Maintenance

PdM is a key part of the industry. Recently, with the development of machine learning, a large number of researchers have introduced machine learning techniques, especially deep learning, into PdM. Different researchers have proposed various deep learning algorithms [6-10] for PdM.

Digital twin is a prevailing topic in smart manufacturing. Ding et al. (2019) [13] proposed a digital twin approach for modelling the remaining useful life (RUL) of shearer key parts. Firstly, the simulation was implemented based on the real-world data to obtain more data such as position, posture, trajectory, which can be used for qualitative analysis. Then, autoencoder and bi-directional gated recurrent unit were used to construct an RUL prediction model based on the data collected from real-world and quantitative analysis.

Lee (2019) [14] proposed an approach to evaluate the reliability of a complex system. In this approach, discrete-time Markov chains were deployed to predict the health state of the components in a complex system. Then a Bayesian network was used for modelling the reliability of the complex system. Another study on PdM for the multi-component system was proposed by Liang et al. (2019) [15] where a model was developed for the PdM of multi-system multi-components networks. In this model, analytical and numerical techniques were combined to optimize the maintenance policy.

Chong Chen and Ying Liu are with the Institute of Mechanical and Manufacturing Engineering, School of Engineering, Cardiff University, Cardiff, UK (corresponding author e-mail: ChenC54@cardiff.ac.uk).

Xianfang Sun is with School of Computer Science and Informatics, Cardiff University, Queen's Buildings, Cardiff CF24 3AA, UK

Carla Di Cairano-Gilfedder is with Applied Research, BT Technology, Ipswich, IP5 3RE, UK

Scott Titmus is with Rivus Solutions (formerly BT Fleet), Solihull, B37 7YN, UK

Furthermore, a genetic algorithm with the agglomerative mutation was adopted to effectively determine the maintenance policy.

Besides machine learning techniques, statistical approaches also have been widely used in PdM. Since the spare part management is a key issue in the industry, Khan et al. (2019) [16] deployed a mathematical programming model to formulate the failure time of a vessel to optimize the spare part management based on sensor data. Meanwhile, the shortest path dynamic programming formulation is deployed to address the polynomial-time complexity. Cox proportional hazard model is a prevailing model in reliability analysis which can process both censored and uncensored data. Verhagen and Boer (2018) [17] used both time-independent and time-dependent Cox proportional hazard model to estimate the reliability of aircraft components based on historical operational and maintenance data. Furthermore, Extreme value analysis and maximum difference analysis were adopted to identify operational factors which are relevant to component failure.

Colone et al. (2019) [18] proposed a machine learning approach to predict failures in wind turbine drive-train components and quantify its utility. Firstly, Naïve Bayes' network and artificial neural network was utilized to predict the failure event. Then, a cost analysis was conducted to reveal the potential economic benefits with the application of the proposed method.

B. The Studies of gcForest

The gcForest has been widely used in image classification. Li et al. (2018) [19] deployed a gcForest to establish a cloths classification model. Liu and Yang (2019) [20] applied a gcForest for visual tracking using unmanned aerial vehicle image sequences. In this study, compressive sensing theory was used to reduce the feature dimension so as to increase the computation speed. Yang et al. (2018) [21] combined a simple region proposal network and a gcForest for ship detection based on thermal remote sensing image, where the region proposal network is designed to pre-process images so as to extract candidate regions from complex backgrounds. Mou et al. (2019) [22] proposed a structured gcForest modelling approach for on-road vehicles behavior prediction. The images of on road automobiles were used as input. The label of the data was first obtained via structure learning, before gcForest was utilized to establish a prediction model.

The gcForest also has been applied in the medical field. Dong et al. (2019) [23] proposed a multi-weighted gcForest algorithm to build a lung adenocarcinoma staging prediction model. The weights of random forests (RF) in the standard gcForest are the same, which might be not reasonable. The authors assign weights by determining the hypervolume under multi-flow of each forest. Furthermore, a sorting optimization algorithm was explored to optimize the multi-grained scanning (MGS) stage of gcForest to further leverage the algorithm performance. Ray (2018) [24] deployed a gcForest for disease classification within dermoscopic images. The images were first processed using the ResNet-50, which is a famous convolutional neural network structure. The features extracted from the ResNet were then fed into the gcForest to build a classifier.

The gcForest also has been investigated in engineering, especially in fault diagnosis. Hu et al. (2018) [25] proposed an approach that combines a deep Boltzmann machine and a multi-grained forest for fault diagnosis. The deep Boltzmann machine was first used to transform the raw data into a binary representation. Then, the gcForest was deployed for modelling based on the pre-processed data. Wang et al (2018) [26] proposed a combined method for fault diagnosis. A feature selection approach based on Spearman's correlation was first deployed to remove redundant features. Then, the k -means algorithm was adopted to determine the label of data, which is the degradation of a machine. Finally, a gcForest was used to build a classifier based on the processed data.

In PdM, Liu et al. (2018) [27] deployed a gcForest for fault recognition of rolling bearing based on the sensor data. Liu et al. (2019) [28] proposed a gcForest-based end-to-end intelligent fault diagnosis method for hydraulic turbine fault diagnosis based on sensor data. In this study, the robustness of gcForest to noise was tested and revealed.

From the literature, it can be seen that the gcForest has been widely used in different fields such as medication and PdM. However, its application in regression mission has not been reported. Our previous study [29] revealed that a deep neural network can be helpful in automobile TBF modelling with GIS data. Owing to the fact that the gcForest does not require a high computational load and large efforts in parameters tuning in comparison with the deep neural network, it can also be beneficial in TBF modelling. Hence, it is worthwhile to introduce the gcForest into TBF modelling and reveal its performance in regression mission.

III. METHODOLOGY

A. Overall Procedure of Automobile Maintenance Modelling

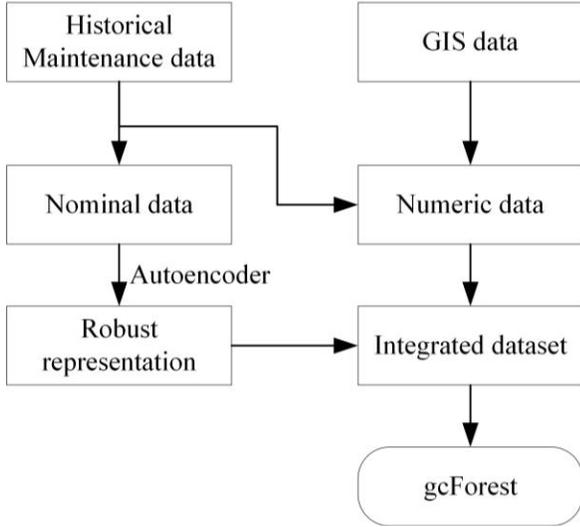
Fleet management can be beneficial from the prediction of TBF to better schedule maintenance actions. In order to establish an automobile TBF prediction model based on historical maintenance data and GIS data, the following methodology was adopted.

Firstly, there are both nominal and numeric data in historical maintenance dataset, and the nominal data needs to be transformed into numeric form. In our previous study [29], one-hot encoding and autoencoder were used to transform the nominal data into numeric representation and the results show its effectiveness. Hence, in this study, the scheme is adopted for nominal data transformation.

Secondly, GIS data is collected from different sources and then summarized using GIS software. The automobiles in the same garage of a fleet management company are assumed to run in the same working area. Using the geographical coordinates of the working area, the GIS data within each area can be captured and summarized.

Thirdly, after the historical maintenance data and GIS data are prepared, both types of data are then concatenated to form a new dataset. In the modelling stage, a gcForest algorithm is deployed to establish a TBF prediction model based on the concatenated data. Figure 1 demonstrates the overall flow of the methodology. The technical details of gcForest are reported in the next section.

Figure 1. The overall flow of the methodology.



B. gcForest

Inspired by deep neural network and ensemble learning, Zhou and Feng (2017) [11] proposed a gcForest scheme to learn the deep feature representation within data. Figure 2 shows the structure of the gcForest. There are two main parts in the gcForest, which are the MGS and the cascade forest. The raw data input contains x dimensions. Different sliding windows with sizes x_1 , x_2 and x_3 are adopted to extract features from the data. The extracted features were fed into two algorithms, which are RF and completely-RF, to further learn the hidden patterns within data. Completely-RF is the RF that the number of maximum features in each node is one, while the number of maximum features in each node in a normal RF is determined by the square root of the number of features.

Another part of the gcForest is the cascade forest. In each layer of the cascade forest, there are two RFs (white) and two completely-random tree forests (blue). As an RF is an ensemble model, each layer in a cascade forest can be deemed as an ensemble of the ensemble. The output of each layer is then concatenated with the original input, which is similar to the mechanism of residual neural network. After the features are processed by all the layers in a cascade, a training result is obtained. Then, the processed features obtained in the first cascade are sent into the second cascade to implement the procedure again. After the increment of the result is lower than the pre-determined threshold, the training process ends. The result obtained in the last cascade is the result of the training process. Different from the neural network which possesses the fixed number of layers, the layers of gcForest are determined by the data size and number of features, which affects the convergence speed.

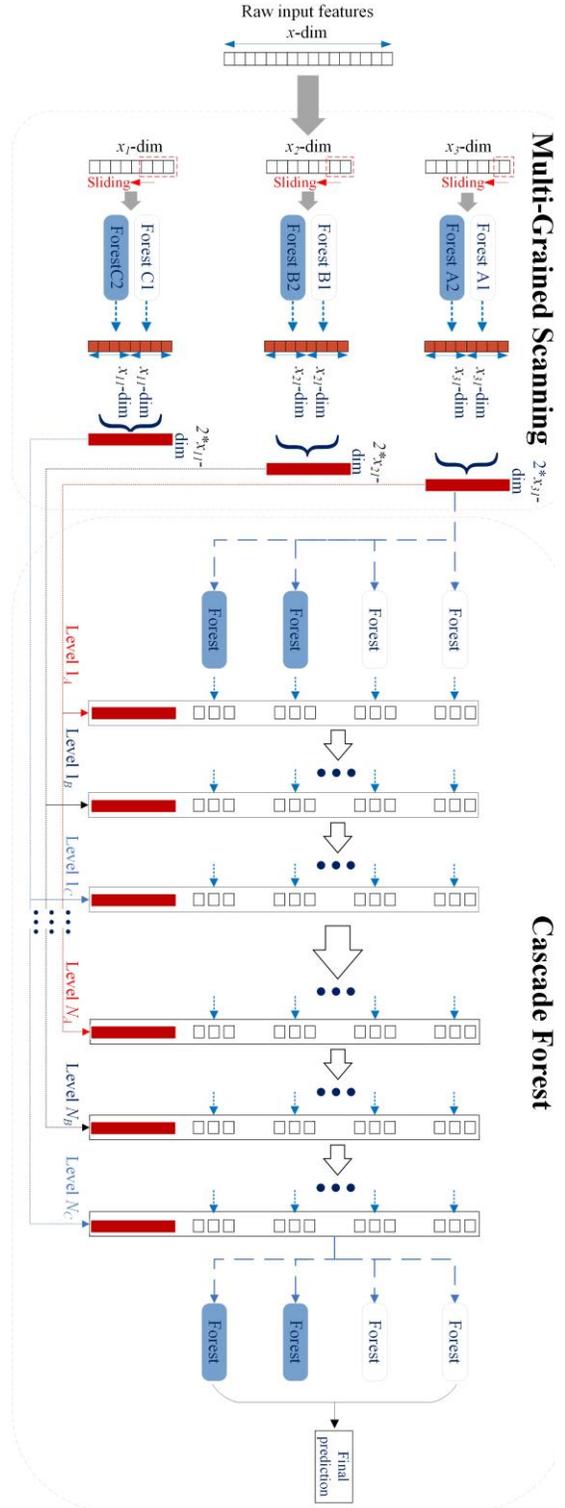
IV. EXPERIMENTAL STUDY

A. Experimental Setup and Data

The data used in this study was provided by a sizeable fleet management company in the UK, which possesses a large number of automobiles all around the UK. In order to set up the experiment, the background of the company needs to be first studied. The automobile managed by this company has

various types of automobiles includes van, personal car and 4 by 4 cars, etc. Currently, run to failure and scheduled check are two main maintenance strategy in this company. Without the information of automobile failure, it is hard to optimize their job scheduling, spare parts management and maintenance planning. Hence, the company under investigation has a strong interest in predicting the failure date of automobiles.

Figure 2. The structure of gcForest.



Since the automobile is a type of repairable asset, an automobile TBF prediction model can be a useful tool for the company. Secondly, the historical maintenance data needs to be collected. The dataset was collected from the garages of the company. When an automobile fails or detected failure during a scheduled check, the maintenance would be scheduled for the automobile. The data relevant to the automobile such as automobile age, mileage and model is recorded at this time. An automobile engine maintenance dataset, which contains over 10 thousand data entries, was collected from the fleet management company. The feature description of historical maintenance data is shown in Table 1.

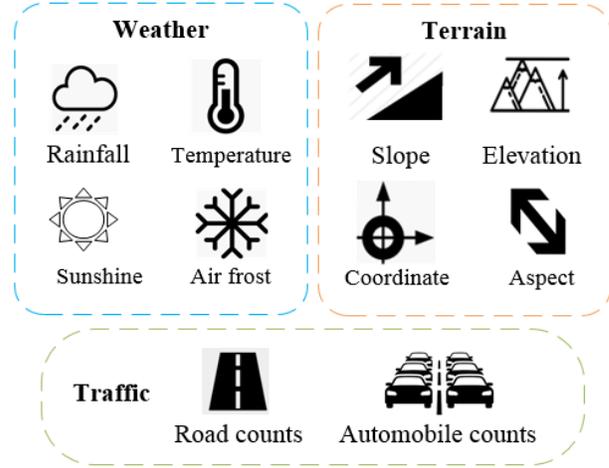
TABLE I. FEATURE DESCRIPTION OF HISTORICAL MAINTENANCE DATA

Features	Note	Features	Note
$nRepair$	The times of engine experienced maintenance	Last TTF	The time to failure of last maintenance
$PAge$	The age of automobile engine	Seq	A time index for automobile
Vage	The age of automobile	$Regions$	Four binary attributes
$CumM$	The cumulative miles when a failure occurs	$Model$ (Nominal)	The model of automobile
$Model_Year$	The year of the automobile model	WS (Nominal)	The workstation of automobile
$Last\ TBF$	The TBF before last maintenance	$Area$ (Nominal)	The area of automobile

Thirdly, the GIS data was collected. The location of each garage is different which means the GIS factors such as weather, terrain and traffic are different. Hence, these GIS factors may impact automobile lifecycle. In order to collect the GIS data of a garage, a circular mobility area was set with a radius of 30 km. The garage was set as the centre of the mobility area. The determination of the radius considered the daily mileage of the automobiles belongs to the same garage.

Three categories of GIS features were collected in this study, including weather, terrain and traffic. The weather data was collected from the Met Office, UK. Since the weather data was collected monthly, its granularity is relatively small. Hence, the mean and standard deviation (Std) of the weather features before the automobile start date is adopted. The terrain data was collected from the ArcGIS software by analyzing the elevation map of the UK. The terrain is relatively stable in a long period. The mean and Std of the terrain data in each mobility area are summarized. The traffic data was collected from the transportation department of the UK. It was collected yearly. The granularity of traffic data is relatively large, and therefore the nearest traffic data before the automobile start date is adopted. After all the GIS data was collected, the data was analyzed and summarized using ArcGIS software. In total, there were 21 GIS features adopted in this study. The categories of GIS features are illustrated in Figure 3.

Figure 3. Categories of features in GIS data.



The three nominal features in the historical maintenance dataset were converted to numeric data using one-hot encoding and autoencoder techniques proposed in [29]. Hence, both techniques were deployed in this study again to convert the nominal data to numeric data. After the nominal data was converted, the maintenance dataset and GIS dataset were concatenated. After concatenation, there are 28 features in the concatenated dataset. The abnormal values in the concatenated dataset were then removed and the dataset was normalized. Then, the abnormal data entries were removed. The TBF of some data entries is too low which were deemed as noisy data. Hence, the data entries with TBF lower than 30 days were removed. After this step, there are over 10 thousand data entries in the dataset. Finally, the dataset was reshuffled to yield comprehensive results.

The algorithm performance of gCForest needs to be evaluated. In this study, two metrics, root mean square error (RMSE) and model correlation coefficient (MCC), were used to evaluate the algorithm performance. The training time of each algorithm was recorded for comparison of the computational cost. Meanwhile, 10-fold cross-validation was implemented to deliver comprehensive results. MCC is used to measure the correlation between two variables. The MCC can be expressed as:

$$MCC = \frac{S_{PA}}{\sqrt{S_P S_A}}, \quad (1)$$

where,

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

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

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.

RMSE is a scale-dependent metric which measures the difference between the prediction values and the actual values. The expression of RMSE is:

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

In this study, a gcForest algorithm was deployed to build an automobile TBF prediction model. In this model, the base estimator was changed from RF classifier to RF regressor and the evaluation metrics in gcForest was changed from classification accuracy to MCC. The parameters were tuned after different trials. The configuration of gcForest is shown in Table 2.

TABLE II. KEY PARAMETERS OF GCFOREST

Parameter	Value	Parameter	Value
Number of RF tree in MGS	201	Number of cascade RF	2
Number of cascade RF tree	201	Number of cascade layer	4
Minimum samples in MGS	0.5%	Minimum samples in cascade forest	0.05%

To show the merits of gcForest, three different algorithms, fully-connected neural network (FCNN), support vector machine (SVM), and random forest (RF), were used as benchmarks. FCNN was developed using Python Keras [30] package, while SVM and RF were developed using Python Sklearn [31] package. The configurations of SVM and RF were set as default. The algorithm performance of FCNN is highly dependent on the neural network size. In order to reveal the computational cost of gcForest, the neural network size was tuned to get a similar algorithm performance of gcForest. The number of hidden layers in FCNN was set as four and the number of neurons in each hidden layer was set as 1000. All tests were conducted on an Intel i5-6500 3.20Ghz PC with 3*8GB RAM. In this study, two experiments were set to reveal the algorithm performance of gcForest in TBF modelling. In the first experiment, four algorithms mentioned above were used to establish the TBF prediction model based on historical maintenance data. In the second experiment, the relationship between data volume and the algorithm performance of gcForest in terms of MCC and RMSE was explored. The data volume of the dataset was adjusted and then fed into gcForest to generate different models. The MCCs and RMSEs were marked and plotted to reveal the relationship.

B. Experimental Results

The modelling results are shown in Table 3. It can be seen that gcForest achieved similar performance in terms of MCC and RMSE in comparison with FCNN. In contrast, SVM achieved the worst performance in terms of MCC and RMSE. The algorithm performance of gcForest and FCNN in terms of MCC and RMSE are better than that of SVM and RF, which indicated that gcForest shows merits in TBF modelling.

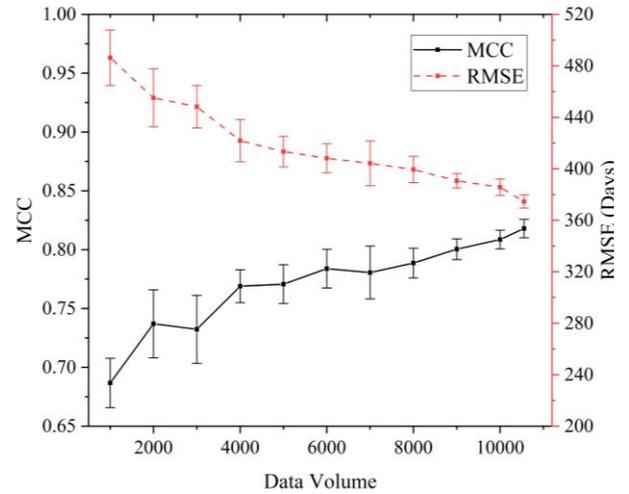
The modelling time reveals the computational cost of an algorithm. It is obvious that both gcForest and FCNN required a longer time to train a model in comparison to the SVM and RF methods, and the modelling time of gcForest is less than that of FCNN.

TABLE III. RESULTS OF MODELLING

Metrics	gcForest	FCNN	SVM	RF
MCC	0.818	0.817	0.759	0.801
RMSE (days)	374.6	375.9	429.5	387.8
Modelling Time (s)	92.34	140.30	5.71	0.08

In order to further reveal the algorithm performance of gcForest, the second experiment was implemented. The data volume was adjusted and yield different modelling results. The results are shown in Figure 4. It can be seen that the algorithm performance of gcForest in terms of MCC first increases dramatically when the data volume rises from 1000 to 5000, following by a steady increase. The algorithm performance of gcForest in terms of RMSE first has an obvious drop and then the decreasing trend tends to be moderate. However, the increasing/ decreasing trend of MCC/ RMSE has not converged in this figure, which suggests that the algorithm performance can be further promoted with the increase of data volume. Meanwhile, it is noticeable that the Std of both metrics shrink along with the growth of data volume. It indicates the algorithm performance in terms of MCC and RMSE tends to be stable when the data volume is large.

Figure 4. The relationship between data volume and the performance of gcForest in terms of MCC and RMSE.



C. Discussion

The experiment results indicated that gcForest can achieve similar performance to FCNN, while it requires less computational cost. That is because gcForest requires fewer parameters in comparison with those neural network-based algorithms. In the era of big data, the computational cost is a big concern as it results in a higher requirement to the hardware. In comparison with FCNN, gcForest shows its merit in modelling time. The relationship between data volume and the performance of gcForest in terms of MCC and RMSE was revealed in this study. This insight can be helpful in the deployment of gcForest in the actual fleet management. In the future, how to further leverage the algorithm performance of gcForest will be investigated.

V. CONCLUSION

Automobile predictive maintenance is important to fleet management. In this paper, the studies of predictive maintenance and gcForest were reviewed. An emerging deep learning algorithm called gcForest and how it can be introduced into automobile predictive maintenance were explained. An experimental study using a large number of automobile historical maintenance data was implemented. The gcForest was first adopted to establish an automobile TBF prediction model. The results indicate that it can achieve similar performance to FCNN's while it requires less computational cost.

REFERENCES

- [1] J. Yan, Y. Meng, L. Lu, and L. Li, "Industrial big data in an industry 4.0 environment: Challenges, schemes, and applications for predictive maintenance," *IEEE Access*, vol. 5, pp. 23484-23491, 2017.
- [2] T. Borgi, A. Hidri, B. Neef, and M. S. Naceur, "Data analytics for predictive maintenance of industrial robots," in *Proceedings of International Conference on Advanced Systems and Electric Technologies, IC_ASET 2017*, 2017, pp. 412-417.
- [3] F. Civerchia, S. Bocchino, C. Salvadori, E. Rossi, L. Maggiani, and M. Petracca, "Industrial Internet of Things monitoring solution for advanced predictive maintenance applications," *Journal of Industrial Information Integration*, vol. 7, pp. 4-12, 2017.
- [4] M. Camero, "An evaluation system of the setting up of predictive maintenance programmes," *Reliability Engineering & System Safety*, vol. 91, no. 8, pp. 945-963, 2006.
- [5] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.
- [6] X. Li, Q. Ding, and J.-Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering & System Safety*, vol. 172, pp. 1-11, 2018/04/01/ 2018.
- [7] X. Xu, Q. Wu, X. Li, and B. Huang, "Dilated Convolution Neural Network for Remaining Useful Life Prediction," *Journal of Computing and Information Science in Engineering*, vol. 20, no. 2, 2020.
- [8] J. Zhang, P. Wang, R. Yan, and R. X. Gao, "Long short-term memory for machine remaining life prediction," *Journal of Manufacturing Systems*, vol. 48, pp. 78-86, 2018/07/01/ 2018.
- [9] A. Al-Dulaimi, S. Zabihi, A. Asif, and A. Mohammed, "NBLSTM: Noisy and Hybrid CNN and BLSTM-based Deep Architecture for Remaining Useful Life Estimation," *Journal of Computing and Information Science in Engineering*, pp. 1-14, 2019.
- [10] G. Sateesh Babu, P. Zhao, and X.-L. Li, "Deep Convolutional Neural Network Based Regression Approach for Estimation of Remaining Useful Life," in *Database Systems for Advanced Applications*, Cham, 2016, pp. 214-228: Springer International Publishing.
- [11] Z.-H. Zhou and J. Feng, "Deep forest," *arXiv preprint arXiv:1702.08835*, 2017.
- [12] A. Rikalovic, I. Cosic, and D. Lazarevic, "GIS based multi-criteria analysis for industrial site selection," *Procedia Engineering*, vol. 69, pp. 1054-1063, 2014.
- [13] H. Ding, L. Yang, and Z. Yang, "A predictive maintenance method for shearer key parts based on qualitative and quantitative analysis of monitoring data," *IEEE Access*, vol. 7, pp. 108684-108702, 2019.
- [14] D. Lee and R. Pan, "Evaluating reliability of complex systems for Predictive maintenance," *arXiv preprint arXiv:1902.03495*, 2019.
- [15] Z. Liang and A. K. Parlikad, "Predictive group maintenance for multi-system multi-component networks," *Reliability Engineering & System Safety*, vol. 195, p. 106704, 2020.
- [16] R. Kian, T. Bektaş, and D. Ouelhadj, "Optimal spare parts management for vessel maintenance scheduling," *Annals of operations research*, vol. 272, no. 1-2, pp. 323-353, 2019.
- [17] W. J. C. Verhagen and L. W. M. De Boer, "Predictive maintenance for aircraft components using proportional hazard models," *Journal of Industrial Information Integration*, vol. 12, pp. 23-30, 2018/12/01/ 2018.
- [18] L. Colone, N. Dimitrov, and D. Straub, "Predictive repair scheduling of wind turbine drive-train components based on machine learning," *Wind Energy*, vol. 22, no. 9, pp. 1230-1242, 2019.
- [19] L. Han, Z. Haihong, Y. Erxin, B. Yuming, and L. Huiying, "A Clothes Classification Method Based on the gcForest," in *2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC)*, 2018, pp. 429-432: IEEE.
- [20] F. Liu and A. Yang, "Application of gcForest to visual tracking using UAV image sequences," *Multimedia Tools and Applications*, vol. 78, no. 19, pp. 27933-27956, 2019.
- [21] F. Yang, Q. Xu, B. Li, and Y. Ji, "Ship detection from thermal remote sensing imagery through region-based deep forest," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 3, pp. 449-453, 2018.
- [22] L. Mou, S. Mao, H. Xie, and Y. Chen, "Structured behaviour prediction of on-road vehicles via deep forest," *Electronics Letters*, vol. 55, no. 8, pp. 452-455, 2019.
- [23] Y. Dong, W. Yang, J. Wang, J. Zhao, and Y. Qiang, "MLW-gcForest: A Multi-Weighted gcForest Model for Cancer Subtype Classification by Methylation Data," *Applied Sciences*, vol. 9, no. 17, p. 3589, 2019.
- [24] S. Ray, "Disease Classification within Dermoscopic Images Using features extracted by ResNet50 and classification through Deep Forest," *arXiv preprint arXiv:1807.05711*, 2018.
- [25] G. Hu, H. Li, Y. Xia, and L. Luo, "A deep Boltzmann machine and multi-grained scanning forest ensemble collaborative method and its application to industrial fault diagnosis," *Computers in Industry*, vol. 100, pp. 287-296, 2018.
- [26] C. Wang, N. Lu, Y. Cheng, and B. Jiang, "Deep forest based multivariate classification for diagnostic health monitoring," in *2018 Chinese Control And Decision Conference (CCDC)*, 2018, pp. 6233-6238: IEEE.
- [27] Q. Liu, H. Gao, Z. You, H. Song, and L. Zhang, "Gcforest-based fault diagnosis method for rolling bearing," in *2018 Prognostics and System Health Management Conference (PHM-Chongqing)*, 2018, pp. 572-577: IEEE.
- [28] X. Liu *et al.*, "Deep forest based intelligent fault diagnosis of hydraulic turbine," *Journal of Mechanical Science and Technology*, vol. 33, no. 5, pp. 2049-2058, 2019.
- [29] C. Chen, Y. Liu, X. Sun, C. Cairano-Gilfedder, and S. Titmus, "Automobile maintenance prediction using deep learning with GIS data," *Procedia CIRP*, vol. 81, pp. -, 2019.
- [30] F. Chollet, "Keras: Deep learning library for theano and tensorflow," *URL: <https://keras.io/k>*, vol. 7, p. 8, 2015.
- [31] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825-2830, 2011.