


# Automobile Predictive Maintenance Using Deep Learning


Sanjit Kumar Dash, College of Engineering and Technology, Bhubaneswar, India

 <https://orcid.org/0000-0003-4244-7591>

Satyam Raj, College of Engineering and Technology, Bhubaneswar, India

Rahul Agarwal, College of Engineering and Technology, Bhubaneswar, India

Jibitesh Mishra, College of Engineering and Technology, Bhubaneswar, India

 <https://orcid.org/0000-0003-2761-470X>

## ABSTRACT

There are three types of maintenance management policies: run to failure (R2F), preventive maintenance (PvM), and predictive maintenance (PdM). In both R2F and PdM, we have the data related to the maintenance cycle. In the case of preventive maintenance (PvM), complete information about maintenance cycle is not available. Among these three maintenance policies, predictive maintenance (PdM) is becoming a very important strategy as it can help us to minimize the repair time and the associated cost with it. In this paper, the authors have proposed PdM, which allows the dynamic decision rules for the maintenance management. PdM is achieved by training the machine learning model with the datasets. It also helps in planning of maintenance schedules. They especially focused on two models that are binary classification and recurrent neural network. In binary classification, they classify whether the data belongs to the failure class or the non-failure class. In binary classification, the number of cycles is entered, and the classification model predicts whether it belongs to the failure or non-failure class.

## KEYWORDS

Binary Classification, Deep Learning, Predictive Maintenance, Recurrent Neural Network

## 1. INTRODUCTION

In this era of technology, ML is playing a vital role in making our life easier. Basically ML is the scientific study of the statistical model that computer system uses to perform specific task without any explicit instruction. It relies on patterns and inference instead. Automobile sector is one such sector which is using ML. In the automobile sector the automobile maintenance has been the major talk in today's scenario. All machines are prone to failure. Automobile maintenance can be carried out by three major maintenance policies (Susto et al., 2012): Run to failure R2F, PvM and PdM. In R2F, the actions are carried out after the failure thus this type of maintenance is least effective as the cost associated is more and downtime gets increased. In PvM, the actions are carried out according to planned schedule. The problem with PvM is that it involves an unnecessary action which leads to the wastage of the resource and increase in cost. In PdM, the actions are carried out based on an estimated health status of a piece of equipment. PdM systems allow advance detection of pending failures. (Krishnamurthy et al., 2005)

DOI: 10.4018/IJAIML.20210701.0a7

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

In automobile maintenance and repair, it is necessary to properly diagnose the cause of failure (Wang et al., 2018) so that one does not needlessly replace or worsen vehicle components. To be able to gather these information PdM can serve as the effective solution (Mobley, 2002). We can use ML for the purpose of PdM in the automobiles. This can help us in reducing associated cost and downtime thus increases the efficiency.

In this paper, we take into consideration PdM and try to develop a model that would allow us to carry out PdM process, to detect fault in the parts of automobiles and to know about the fault before its occurrence and take precautions at the earliest before breakdown of the automobile. The model will identify process and logistics variables collected during production to identify the reasons of the degradation of the automobile. The prediction of machine health can, not only significantly reduce the unexploited downtime and expensive labour costs but also ensures safe operation and optimizes the maintenance plan. We have used two ML techniques in the proposed model i.e. Binary classification and Regression. Binary classification tells about whether Fault will be encountered in the machine parts and through regression the cycles or epoch after which the fault happens will be predicted (Ali & Zain, 2019; Costello et al., 2017; Liu et al., 2015; Sherstinsky, 2020). We have also used LSTM under Regression as it can process long sequences. LSTM networks apply well to PdM domain since they are good at learning from a specific sequence. The model developed will be trained. Then later the results of the two techniques will be analysed. The results are plotted in graphs showing the credibility of the model. The fault detection of the degraded automobile part will help to solve our problem of maintenance.

PdM solves the challenges posed by Run to Failure and preventive maintenance. A model to implement PdM at the larger level is required. The model should be able to predict the failure occurrence at the right time so as to make use of the resources available effectively. The model will gather information about the health status of the machine parts, process out calculations through periodic cycles and provide effective solution for the maintenance and the faults will be diagnosed and corrected. According to the results gathered through the model, a detailed plan of action can be formulated. The model will make use of the ML techniques in order to provide the valid results.

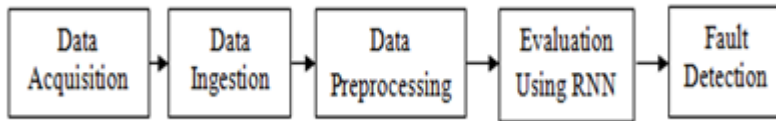
## 2. RELATED WORK

A multiple classifier methodology using ML techniques for PdM is bestowed (Susto et al., 2014). PdM may be a distinguished strategy for handling maintenance problems given the increasing cost to minimize time period and associated prices. The challenge with PdM is generating the “health factors or quantitative indicators or of the critical understanding of a system related to a given maintenance issue, and their relationship with cost of operating and failure risk. The PdM methodology (Langone et al., 2013) permits propulsive decision rules to be adopted for maintenance management, and might be used with high-dimensional and expurgated information issues. This is often achieved by coaching multiple classification modules with totally different prediction horizons to supply different performance trade offspin in terms of frequency of surprising breaks and faulty life, and so using this information in operation cost-based maintenance call system to reduce expected costs.

PdM also supports data-driven strategies (Zhang et al., 2019) that can deal with sensible manufacturing and industrial huge information, particularly for acting health perception (e.g., fault diagnosis and remaining life assessment) (Farrar & Worden, 2012). It tends to classify the precise industrial applications supporting six algorithms of ML (Cline et al., 2017) and deep learning (DL), and compare five performance metrics for every classification. The accuracy (a metric to judge the formula performance) of those PdM applications is analyzed.

ML is the major success that considers the continued digital transformation across industries (Bishop, 2006). ML is employed in several domains. Image analysis is used to spot distinct forms and shapes, such as for medical analyses or face and fingerprint recognition. Deep learning (Alpaydin, 2020) is to come up with rules for knowledge analytics and massive knowledge handling.

Figure 1. System Model



The essential data sources that must be found while thinking of a suitable problem for a PdM (Lu et al., 2009) solution are: The training data should contain of examples about normal operating scheme of the automobile part, along with the failure scheme. Maintenance or repair history containing details about replaced components needs to be known and also the condition of the automobile part which will provide a pattern of the performance of the part with varying time. The ML algorithms like binary classification, regression and multiple classifiers are the choice for carrying out predictive maintenance. Binary classification helps to estimate the probability of the failure of a machine part (Liu et al., 2010). The probability will tell whether the part failed or it can still work. Regression will help to calculate the remaining useful life of the machine part and tell the estimated time of the failure of the parts. Multi class classification helps to determine root cause of a given failure and help suggest the main maintenance actions to be performed to repair a fault (Paolanti et al., 2018).

### 3. SYSTEM MODEL

In the proposed work we developed a system that tells whether automobile engine will have any failure in the future or not. The system model is shown in figure1. Objective of our model is to produce a system suitable for carrying out the maintenance related solution for the automobiles and minimize the operational cost. System model encompasses multiple steps to achieve the purpose. The foremost of them is data acquisition. Such a step includes importing of the requisite packages like numpy, pandas, matplotlib, keras and inclusion of dataset into the model. The subsequent step that follows requires inclusion of 21 parameters of the automobile engine from the target dataset. These parameters are chosen as per their significance to the end goal. The dataset is then presented as input to the ML model. This embarks the process of data ingestion. It is followed by data preprocessing where the dataset is transformed into a format which the model comprehends. This crucial phase performs handling of missing data and feature scaling using MinMax Normalization. Once the dataset is ready to use it is trained for multiple epochs with the dataset. Once training of the mode is complete, the model is validated against test input and accuracy is checked. This final model helped predict whether the given data belonged to failure class or the non-failure class thus enabling classification. Thus, it helps to predict possible failure in advance.

### 4. DEEP LEARNING IN PREDICTIVE MAINTENANCE

The proposed ML model is being developed using Binary Classification and Recurrent Neural Network. Binary Classification just gives the information whether there will be any failure or not. In case of Recurrent Neural Network it gives the exact information regarding failure.

#### 4.1 Binary Classification

Binary classification technique is used to determine and assign two different categories to an entity. The categories are referred as labels which are generally in the form of 0 and 1. In the proposed work, the labels are categorized into F and NF. F refers to Faulty and NF refers to Non-Faulty. The F here

indicates that a particular machine part is declared faulty and on which the maintenance activity will be carried out. The NF indicates that the particular machine part does not require to be repaired.

The MinMax normalization is used in data preprocessing step for scaling. There are two layers of LSTM used. The loss function is calculated by binary\_crossentropy. Optimizer is *adam* and the activation function is *sigmoid*, since it is a binary classification problem.

$$X_{new} = \frac{X_i - \min(x)}{\max(x) - \min(X)} \quad (1)$$

The above equation performs min max normalization.

$$y = \zeta(x) = \frac{1}{1 + e^{-z}} \quad (2)$$

Confusion matrix is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known. The results of the binary classification are evaluated and the confusion matrix is formed. It gives us insight not only into the errors being made by a classifier but more importantly the types of errors that are being made. The performance of the binary classifier is calculated by various metrics like recall, accuracy and precision. Precision and recall are calculated through the values of the four parameters of confusion matrix.

## 4.2 Recurrent Neural Network

A recurrent neural network (RNN) is a class of artificial neural networks where the output from the previous step is used as the input in the present step. In traditional recurrent neural network input and output are independent of each other however there are certain situations where in order to predict next sequence we need to remember about the previous sequence in cases like that RNN proves to be useful. This is possible because of the hidden layer which remembers about the sequence.

The Min Max normalization is used in data preprocessing step for scaling purpose. We build a deep network. The first layer is an LSTM (Long Short-term Memory) layer with 100 units and second layer with 50 units. Dropout is used after each layer to prevent overfitting. The dropout has a value of 0.2 for both the layers. The loss function is calculated by Mean squared error (MSE). Optimizer is *rmsprop* and the activation function is linear since it is a regression problem.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \tilde{y}_i)^2 \quad (3)$$

n is the number of data points

- $y_i$  is the observed value
- $\tilde{y}_i$  is the predicted value

RNN uses LSTM which remembers all the information of the past and helps to make accurate prediction. In RNN we have one input layer and one output layer and number of hidden layer is not fixed we can have any number of the hidden layer in between input and output layer. RNN provides same weights and biases to all the layers which convert the independent activation into dependent

activation which helps in reducing the complexity. The input, output and hidden layers are combined such that weights and biases of the hidden layers are same.

## 5. RESULTS

The model is trained using Engine Degradation Simulation Dataset .The packages used in the model are numpy, pandas, matplotlib and keras. Each data has 21 parameters named as s1,s2 upto s21. Setting1, Setting2 and Setting3 tells about the configuration of the sensors. In both the models the first step is Data Ingestion where the data is brought to the coding environment then data is preprocessed. After the data is processeed both classification and the regression model is trained. The results of the model evaluation is shown in this section.

### 5.1 Result using Binary Classification

During the training of the classification model Sigmoid function was used as the activation function. Sigmoid function was so that our values is in the range of 0 to 1. It is a differentiable function as result of which we can find its slope between any two points. Sigmoid function is monotonic but the function derivative is not.It is especially used for models where we have to predict the probability as an output. Loss function used in the binary classification is Binary cross entropy. It computes the loss between true and predicted values. Optimizer used for the same was Adam. The model is trained with 100 epochs with batch size of 200. The classification is done taking into consideration the classifiers as F(Faulty) and NF(Non Faulty) The F describes the faults which does happen in the parts of automobiles and NF describes the non faulty parts which will work just fine.

The result in Table 1 provides the output generated when we trained data according to binary classification. The table describes the epoch cycle, the corresponding loss value of the function generated at each phase and the accuracy with which the data has been trained. The accuracy tells us the success of our training data. Validation loss (val\_loss) is the loss value of the function generated at the validation phase . The Validation accuracy (val\_acc) is the accuracy of the model that we get by testing the model against the validation data.

The above table shows the results of binary classification represented by the parameters loss, accuracy value, and validation loss and validation accuracy. The results shown are for the first 10 epochs. In testing, the model is checked and evaluated against the training data and generates confusion matrix as given below:

$$\begin{bmatrix} 12345 & 186 \\ 86 & 3014 \end{bmatrix}$$

In the confusion matrix x-axis represents true labels and y-axis represents predicted labels. Further we can use the confusion matrix to calculate accuracy, precision and recall value using the equations given below.

$$\text{Accuracy} = \frac{\text{TruePositive} + \text{TrueNegative}}{\text{TruePositive} + \text{TrueNegative} + \text{FalsePositive} + \text{FalseNegative}} \quad (4)$$

$$\text{Precision} = \frac{\text{TruePositive}}{\text{Truepositive} + \text{FalsePositive}} \quad (5)$$

Table 1. Results of classification

Epoch	loss	acc	val_loss	val_acc
1	0.2637	0.891	0.117	0.9476
2	0.1011	0.9603	0.1361	0.9373
3	0.0813	0.9663	0.656	0.9757
4	0.0734	0.97	0.0324	0.8872
5	0.0691	0.9701	0.0373	0.9923
6	0.0644	0.9725	0.0698	0.9744
7	0.0652	0.9729	0.0416	0.9847
8	0.0516	0.9782	0.0365	0.9859
9	0.0547	0.9763	0.0366	0.9859
10	0.0803	0.9655	0.0473	0.9783

$$\text{Recall} = \frac{\text{TruePositive}}{\text{TruePositive} + \text{FalseNegative}} \quad (6)$$

By using equation 5 & 6, precision and recall values are calculated as 0.941875 and 0.9722580 respectively. Further using equation 4, accuracy of validation set is obtained as 98.25%.

Figure 2 shows the results obtained using binary classifications. From the figure, it can be seen that in the first epoch the accuracy of the training data was around 89%. However with increasing number of epochs it can be clearly seen that accuracy has significantly increased and after 12 epochs it is becoming almost constant. The same can be said for the test data. The accuracy is indicating the correctness of the model.

The validation dataset provides an unbiased evaluation of a model fit on the training dataset while tuning the model's hyper-parameters. This phase will make us validate the results considered at the previous phases. The data is considered against the practical data which we receive through the sensors which will be connected to each part of the automobile. In validation, the model is checked and evaluated against the training data and generates confusion matrix as given below:

$$\begin{bmatrix} 67 & 1 \\ 3 & 22 \end{bmatrix}$$

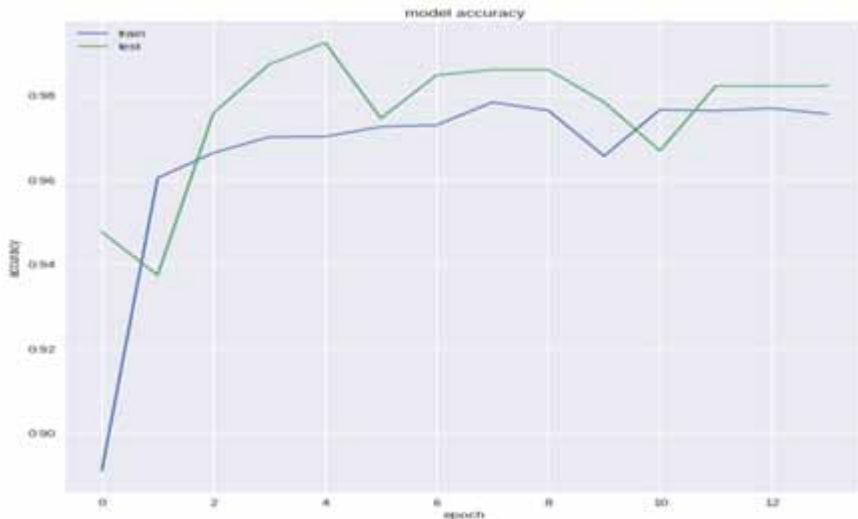
By using equation 5 & 6, precision and recall values are calculated as 0.95652 and 0.88 respectively. The F1 score conveys the balance between the precision and the recall and can be calculated using the equation given below.

$$F1 = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (7)$$

Evaluating equation 7, we get the value of F1-score as 0.91666. Further using equation 4, accuracy of validation set is obtained as 95.69%.

Figure 3 depicts the prediction value graph for the binary classification model. Its value can be either 0 or 1. The predicted and actual data comparison is done using the graph. The results show that there is not much difference in the values which will provide a good choice for the model.

Figure 2. Accuracy vs epoch



## 5.2 Results using Recurrent Neural Network

Regression process provides the better solution for PdM of automobiles. The regression process based model developed in the project helps to predict the fault that will be encountered in different parts of the automobiles. It also helps predict the time after which there will be degradation in the working of parts so as to carry out maintenance according to the results. RNN takes into consideration the results achieved for one cycle of the particular automobile part and use it to track the working

Figure 3. Prediction values graph

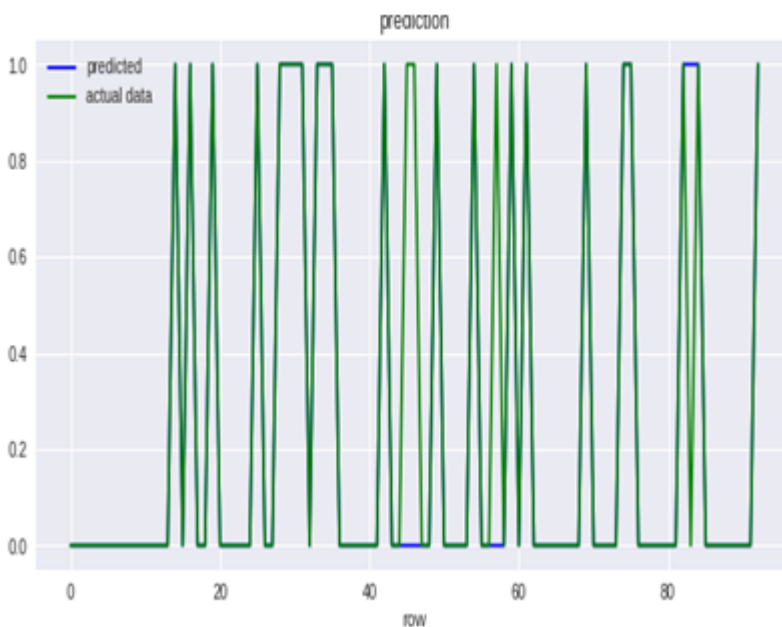


Table 2. Result of Regression

loss	Mean_absolute_error	r2_keras	val_loss	val_mean_absolute_error	val_r2_keras
8710.001	74.5754	-1.69E+00	8121.8596	71.6617	-2.44E+00
8034.3115	70.6539	-1.48E+00	7602.9425	68.65	-2.19E+00
7524.771	67.7182	-1.32E+00	7112.1014	65.8176	-1.96E+00
7048.6777	64.9854	-1.17E+00	6652.79	63.1844	-1.75E+00
6601.8182	62.494	-1.03E+00	6219.2346	60.7198	-1.55E+00
6175.6531	60.0687	-9.02E-01	5815.9571	58.4472	-1.36E+00
5789.4928	57.9402	-7.83E-01	5445.6075	56.3806	-1.19E+00
5433.2812	55.9649	-6.71E-01	5097.9849	54.4635	-1.04E+00
5087.7394	54.1268	-5.64E-01	4781.7566	52.7457	-9.03E-01
4782.8234	52.4782	-4.69E-01	4493.8863	51.2095	-7.81E-01

of the part for the next cycles. This process is generally concerned with training the data to provide satisfactory results.

In RNN we have developed a deep network with 2 different LSTM layers .The first layer is a LSTM layer with 100 units followed by another LSTM layer with 50 units. Activation function used in RNN is linear. It is the linear because RNN produces continuous value. Loss function used in RNN is Mean Squared Error and optimiser used in it is RMSProp. It computes the mean of squares of errors between true and predicted value. Dropout is applied after each LSTM layer to control overfitting. Final layer is dense output layer with single unit and linear activation. The working and results obtained is based on recurrent neural network. The below table shows results obtained like loss value, mean absolute error and r2\_keras which is the result of the different layers for each part.

Table 2 shows the results of the regression process for the model. It is represented through the different parameters, loss, mean absolute error, value of R<sup>2</sup> and the associated loss with it. We obtain values for overall MAE for the test data set and the R<sup>2</sup> value as 13.97660 and 0.791272 respectively. The result is plotted in figure 4.

The value R<sup>2</sup> is a statistical measure of how close the data are to the fitted regression line. The value R<sup>2</sup> on initial level for the test and train data was negative. Then on further training of the model the values show rise and reached about 0.79. This indicated that the model is accurately determining the results.

Figure 5 depicts the MAE vs Epoch graph. Mean Absolute Error (MAE) refers to the results of measuring the difference between the actual and predicted values. MAE value for initial epoch was high for test and train data suggesting low accuracy. The subsequent epoch shows the value being degraded to a low indicating the high accuracy. The validation dataset provides an unbiased evaluation of a model fit on the training dataset while tuning the model's hyper-parameters. This phase validated the results considered at the previous phases. The data is considered against the practical data which we receive through the sensors which will be connected to each part of the automobile.

In figure 6, the actual and predicted data is being compared showing the difference providing validation of our results. The blue line represents predicted data and the green line represents actual data which ultimately shows that the model is giving suitable predicted value.



Figure 4. Graph-  $R^2$  vs epoch

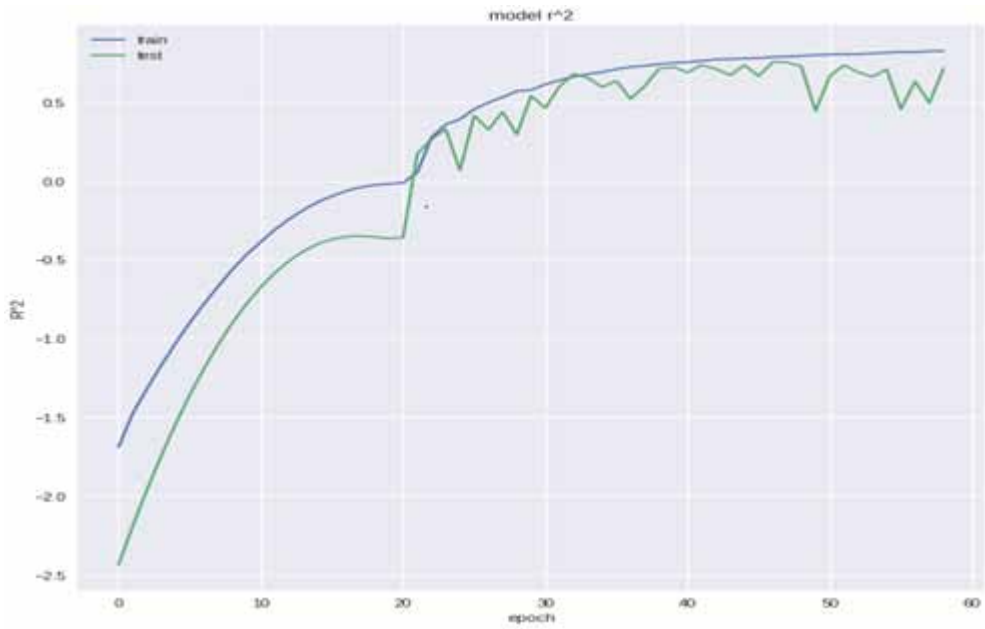


Figure 5. Graph- MAE vs Epoch

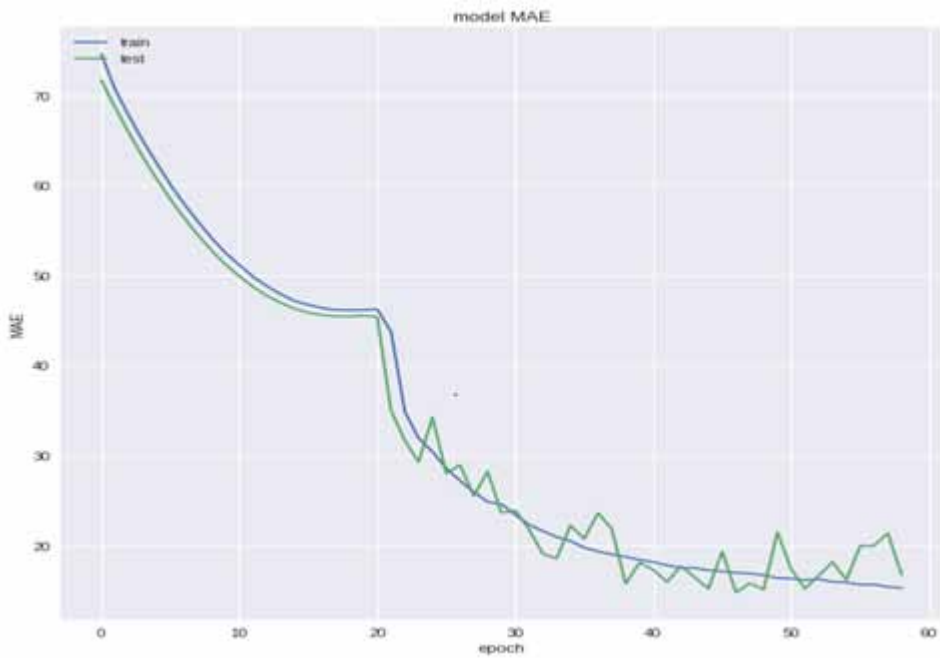
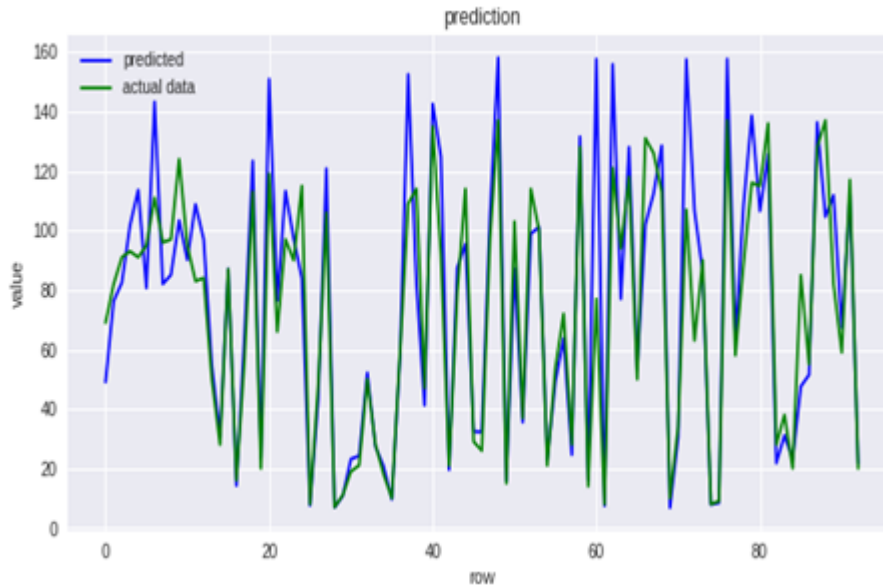


Figure 6. Prediction values graph



## 6. CONCLUSION

In this paper, we solved the problems present in the current implementation of maintenance for automobiles. The problem is that the maintenance related solution prevailing at present had to be done with increased cost and unnecessary breakdowns occur with unplanned downtime. We have used ML techniques to increase effective protection and tell about the future failure of automobile parts. The proposed methodology guarantees improved maintenance management decisions in terms of minimizing operating cost and can be applied to any maintenance problems. RNN reduces the complexity of parameters, unlike other neural networks. Encoding of data is done using label encoding and one-hot encoding. Min-max normalization is applied to configure and standardise the data collected through sensor attach to the parts. The proposed PdM methodology allows dynamical decision rules to be adopted for maintenance management, and can be used with high-dimensional and censored data problems.

## REFERENCES

- Ali, A., & Zain, M. (2019). *A Deep Level Understanding of Recurrent Neural Network & LSTM with Practical Implementation in Keras & Tensorflow*. Recurrent Neural Network & LSTM.
- Alpaydin, E. (2020). *Introduction to machine learning*. MIT press.
- Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.
- Cline, B., Niculescu, R. S., Huffman, D., & Deckel, B. (2017, January). Predictive maintenance applications for machine learning. In *2017 Annual Reliability and Maintainability Symposium (RAMS)* (pp. 1-7). IEEE.
- Costello, J. J., West, G. M., & McArthur, S. D. (2017). Machine learning model for event-based prognostics in gas circulator condition monitoring. *IEEE Transactions on Reliability*, 66(4), 1048–1057. doi:10.1109/TR.2017.2727489
- Farrar, C. R., & Worden, K. (2012). *Structural health monitoring: a machine learning perspective*. John Wiley & Sons. doi:10.1002/9781118443118
- Krishnamurthy, L., Adler, R., Buonadonna, P., Chhabra, J., Flanigan, M., Kushalnagar, N., & Yarvis, M. et al. (2005, November). Design and deployment of industrial sensor networks: experiences from a semiconductor plant and the north sea. In *Proceedings of the 3rd international conference on Embedded networked sensor systems* (pp. 64-75). doi:10.1145/1098918.1098926
- Langone, R., Alzate, C., De Ketelaere, B., & Suykens, J. A. (2013, April). Kernel spectral clustering for predicting maintenance of industrial machines. In *2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)* (pp. 39-45). IEEE doi:10.1109/CIDM.2013.6597215
- Liu, T., Jiang, H., & Wen, J. (2010, October). A binary classification method based on class space model. In *2010 International Conference on Intelligent Computing and Integrated Systems* (pp. 807-809). IEEE. doi:10.1109/ICISS.2010.5657095
- Liu, X., Gao, J., He, X., Deng, L., Duh, K., & Wang, Y. Y. (2015). *Representation learning using multi-task deep neural networks for semantic classification and information retrieval*. Academic Press.
- Lu, B., Durocher, D. B., & Stemper, P. (2009). Predictive maintenance techniques. *IEEE Industry Applications Magazine*, 15(6), 52–60. doi:10.1109/MIAS.2009.934444
- Mobley, R. K. (2002). *An introduction to predictive maintenance*. Elsevier.
- Paolanti, M., Romeo, L., Felicetti, A., Mancini, A., Frontoni, E., & Loncarski, J. (2018, July). Machine learning approach for predictive maintenance in industry 4.0. In *2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)* (pp. 1-6). IEEE.
- Saxena, A., & Goebel, K. (2008). *Turbofan engine degradation simulation data set*. NASA Ames Prognostics Data Repository.
- Sherstinsky, A. (2020). Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D. Nonlinear Phenomena*, 404, 132306. doi:10.1016/j.physd.2019.132306
- Susto, G. A., Beghi, A., & De Luca, C. (2012). A predictive maintenance system for epitaxy processes based on filtering and prediction techniques. *IEEE Transactions on Semiconductor Manufacturing*, 25(4), 638–649. doi:10.1109/TSM.2012.2209131
- Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., & Beghi, A. (2014). Machine learning for predictive maintenance: A multiple classifier approach. *IEEE Transactions on Industrial Informatics*, 11(3), 812–820. doi:10.1109/TII.2014.2349359
- Wang, Y., Wei, Z., & Yang, J. (2018). Feature trend extraction and adaptive density peaks search for intelligent fault diagnosis of machines. *IEEE Transactions on Industrial Informatics*, 15(1), 105–115. doi:10.1109/TII.2018.2810226
- Zhang, W., Yang, D., & Wang, H. (2019). Data-driven methods for predictive maintenance of industrial equipment: A survey. *IEEE Systems Journal*, 13(3), 2213–2227. doi:10.1109/JSYST.2019.2905565

*Sanjit Kumar Dash is working as a Faculty in Information Technology at College of Engineering and Technology, Bhubaneswar, Odisha. He has completed his MTech from Siksha O Anusandhan University, Odisha and PhD from Utkal University, Odisha. His research interests include Wireless Mobile Network, Wireless Sensor Network, Machine Learning.*

*Satyam Raj is currently working as a Software Development Associate in Accenture, Bangalore after graduating from College of Engineering and Technology, Bhubaneswar. His research interest includes Machine Learning and Cloud Computing.*

*Rahul Agarwal is currently working as a Software Development Associate in Accenture, Hyderabad after graduating from College of Engineering and Technology, Bhubaneswar. His research interest includes Machine Learning and Cloud Computing.*

*Jibitesh Mishra, PhD., is working as an Associate Professor in Dept of Computer Science and Application of College at Engineering and Technology, Bhubaneswar, Odisha. His research interests include Mobile Computing, Web Engineering, Fractal Graphics and Machine Learning.*