



IJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: III Month of publication: March 2023

DOI: <https://doi.org/10.22214/ijraset.2023.49878>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Improved RUL Predictions of Aero- Engines by Hyper-Parameter Optimization of LSTM Neural Network Models.

Dr. Shailesh S. Kadre¹, Rohit Gold Paspula²

Cyient Ltd, Hyderabad, Telangana, India

Abstract: *In the industry domains such as transportation, power generation and heavy engineering industries utilize complex machinery, as their working principles are very complex and it is very difficult to predict their exact life and operational availability due to difficulties involved in their operations. PHM- Prognostics and health management is one such domain which supports in the management of this machinery also known as assets for continuous monitoring for their best utilization and maximum efficiency. For any machine, the current health of the system can be determined by operational and raw sensor data. This data can also be utilized for the prediction of remaining useful life (RUL) of the system provided a reliable prediction model is available. It is always be challenging to determine the remaining useful life of complex machines such as turbofan jet engines from the knowledge of the data available from operating conditions and sensor data. An HPT LSTM- advanced hyper parameter tuned LSTM- Long Short Term Memory neural network model is developed to accurately determine the remaining useful life of the turbojet fan engines. To test this predictive model, a NASA developed database CMAPSS- Commercial Modular Aero Propulsion System Simulation is used. The performance of this HPT LSTM is found to be much improved as compared to reference machine learning algorithms such as simple linear regression and Lagged Multi- Layer Perceptron Models*

Index Terms: *Machine Learning, Predictive Maintenance, Remaining Useful Life (RUL), Data Science, LSTM, MLP*

I. INTRODUCTION

In the industry domains such as transportation, power generation and heavy engineering domain utilize complex machines. Some of the biggest challenge related to their effective working is maximizing their operational efficiency, reduced down time, lower cost of maintenance and enhanced safety and reliability. PHM- Prognostics and Health Management plays a very vital role in this domain. Now a days, all these modern machineries are equipped with sensors which provides operational and other critical data related to operations of these machines. This data is historically available and PHM uses them for the predictions of the remaining useful life (RUL) before the catastrophic failure occurs in them.

RUL is a term which is very widely used in the field of PHM which provides significant information about the time to failure data. With the advancement in the machine learning (ML) techniques, it is possible to predict the RUL on the basis of historical sensor and operational data. These techniques are popularly known as data driven approach as they utilize past historical sensor data to train the ML algorithms.

There are some challenges for the historical data based techniques to build reliable prediction model. The relationship between the RUL and monitored/ historical data is very complex. Prior domain knowledge may be required to interpret the historical operational and sensor data to identify the failure mode and to determine their dependencies on the remaining life of the machines. The life of the machine may be a function of a large number of features present in the data which makes the model a large dimensional problem.

In the present research attempts to address the above mentioned challenges in the following way:

- 1) The accuracy of proposed method is improved with respect to accurate prediction of remaining useful life. This is based on the comparison is made with the reference work.
- 2) The developed models are based on the historical data and do not require many domain expertise for the life estimations.
- 3) A single model is capable of predicting the RULs of multiple engine in the available dataset,

NASA has developed a dataset containing run to failure dataset for various turbojet fan engines for multiple engines by the simulated data of Commercial Modular Aero-Propulsion System Simulation module (CMAPSS).

The rest of the paper is organized as follows. Section II provides the brief literature review of the previous work performed in relation to PHM and RUL predictions. The details of LSTM model and CMAPSS dataset are provided in Section III. Section IV presents the step by step procedure to develop the LSTM model for the given dataset. The details of hyper tuning of developed LSTM model by random grid search method are given in section V. The comparison of the results of present research with reference work along with the discussion of key results are presented in section VI. The final conclusion of the study and future work is discussed in Section VII.

II. LITERATURE REVIEW

Staudemeyer et. al. [1] has explained the detailed procedure to develop the long short term memory – Recurrent Neural Network (LSTM- RNN) models. The details of LSTM- RNN are well documented in this study. Duarte et. al. [2] applied the operating condition based NN based techniques on Turbofan aircraft engines.

In the related work by Koen [3] and [4], MLP and LSTM based techniques are applied on the turbojet fan air craft engine on NASA developed CMAPSS datasets [5] . These approaches are mainly data driven and does not require prior domain knowledge of engine failure modes and their correlation with operating and sensor data. In this study the baseline study was performed by considering the Simple linear regression and MLP based models.

Reference [5] provides the details of NASA developed CMAPSS turbojet fan engine dataset which is used for the validation of the developed model in the present research work. Thomare et. al. [6] have developed a LSTM Neural Network and Vector Auto-Regression Models on the CMAPSS database. In this study, they have presented a detailed method to build the LSTM model. Reference [7] to [13] provides the details of various Python libraries used to develop these models.

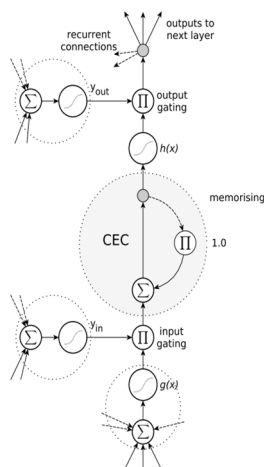


Figure 1: A schematic memory block diagram of typical LSTM model [1].

III. DETAILS OF LONG SHORT TERM MEMORY MODEL- LSTM

The detail of discussion of the LSTM neural network model is out of scope of the current discussion. For the completeness, Figure 1 represents a typical memory block of LSTM cell [1].

The selection of LSTM algorithm for the current research is based on its ability of the ease of work with the time series data and ultimately converting them into sequences.

Data Set: NASA has developed a very well-known dataset [5] Commercial Modular Aero-Propulsion System Simulation (CMAPSS) to test the accuracy and robust ness of various predictions models related to PHM system The sensor data is in the form of time series data which contains the following information:

- 1) Engine Unit Number
- 2) Current operating cycle, which is related to remaining useful life (RUS)
- 3) Operational settings of the engine (3 numbers)
- 4) Sensor Measurements (21 numbers)

Following dataset is used for the current study

- Data Set: FD002
- Train trajectories: 260
- Test trajectories: 259
- Conditions: SIX
- Fault Modes: ONE (HPC Degradation)

Each sensor data has time series related to the some attributes related to the of the engine at various locations. With the increase in number of cycles of engine may develop some faults and system failure which is captured in this sensor data. In the training data set, the time series data is available from the beginning to the end of life of engine. But in test data set, the data is available before the end of engine life and still some remaining useful life if available for that engine. The main objective of this study is to accurately predict this RUL for the test data sets in the form of operational cycles.

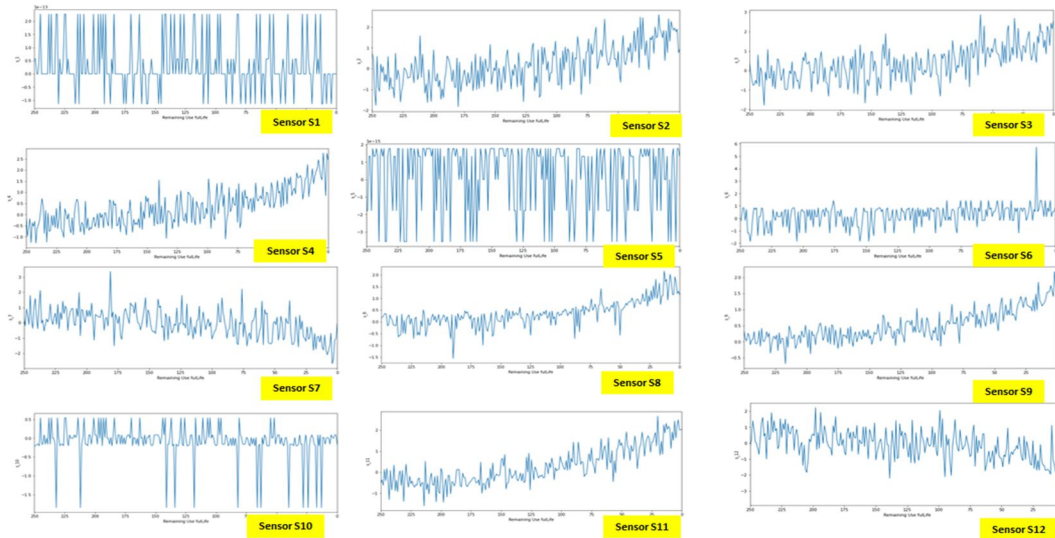


Figure 2: Sensor Data Plot of Sensors S1 to S12 vs Engine RUL for trend analysis for Engine No 2

IV. LSTM MODEL DEVELOPMENT

Following steps are followed for the data preparation.

- 1) *Data Plotting*: Data plotting of each signal with RUL and determine whether each signal shows any trend. Signals with constant and zero values with respect to RUL are straight away dropped.

From the study of Figure 2 and Figure 3, it can be observed that Sensors 's_1', 's_5', 's_6', 's_10', 's_16', 's_18', 's_19' can be dropped.

The remaining Sensors are 's_2', 's_3', 's_4', 's_7', 's_8', 's_9', 's_11', 's_12', 's_13', 's_14', 's_15', 's_17', 's_20', and 's_21'

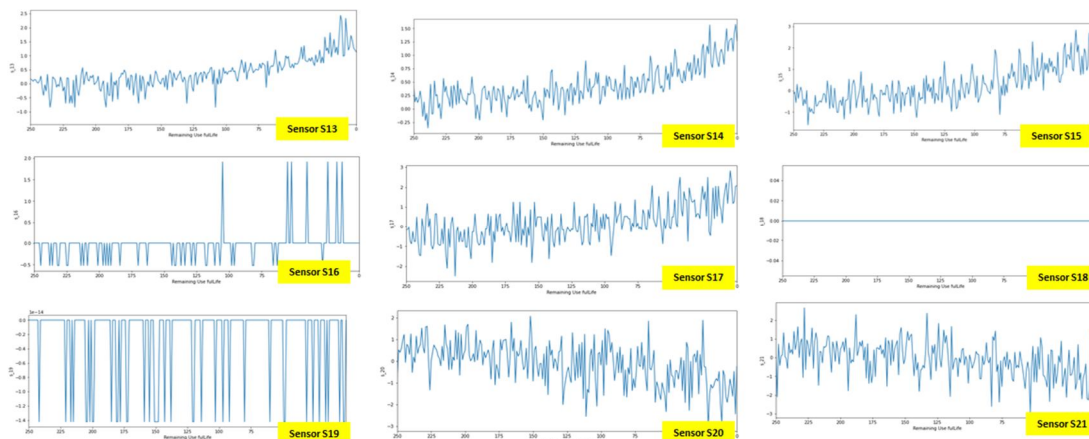


Figure 3: Sensor Data Plot of Sensors S13 to S21 vs Engine RUL for trend analysis for Engine No 2

- 2) *Normalization*: converting settings to string and concatenating makes the operating condition into categorical variables. For Sensor data, Max_min and standard scalar techniques are used
- 3) *Smoothing of Data*: Exponential smoothing [2] is used for noise reduction and smoothing of the signal data with value of alpha = 0.4. Following Figure 4 shows the sample plots for sample sensor data before and after exponential smoothing. From this plot, it can be observed that there is considerable noise reduction after exponential smoothing operation.

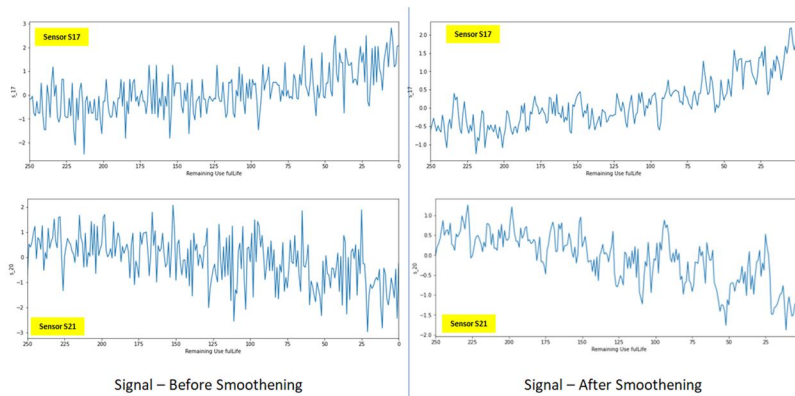


Figure 4: Sample Sensor Data plot for S17 and S21 for Engine No 2

4) *Sequence*

This is a custom made function taken from reference [3] by Ken. Following care is taken for creating the sequence:

- a) Unit numbers should be accounted. So a sequence should be designed in such a way that that that particular sequence should have data pertaining to that unit only. The mixing of data from one engine to others will have error in predicting the correct RUL
- b) Time series of X_t points should be used to estimate the point Y_t not Y_{t+1} .

Following example shows the implementation of this concept:

Table 1: Time series data raw data for various units for single attribute

Sr. No.	Unit Number	X	Y
0	4	40.0	4.6
1	4	40.1	4.5
2	4	40.2	4.4
3	4	40.3	4.3
4	4	40.4	4.2
5	5	50.0	5.6
6	5	50.1	5.5
7	5	50.2	5.4
8	5	50.3	5.3
9	5	50.4	5.2

For the sequence length of 4, following sequences are generated

Table 2: Sequence generation for training LSTM model from time series data

Sequence 1	Prediction label	Sequence 2	Prediction label	Sequence 3	Prediction label	Sequence 4	Prediction label
40.0		40.1		50.0		50.0	
40.1		40.2		50.1		50.1	
40.2		40.3		50.2		50.2	
40.3	4.3	40.4	4.2	50.3	5.3	50.3	5.2

A. Predictions with LSTM approach

Following image gives the parameters used to create baseline LSTM design:

```
Model: "sequential"
-----
```

Layer (type)	Output Shape	Param #
masking (Masking)	(None, 20, 14)	0
lstm (LSTM)	(None, 20, 256)	277504
dropout (Dropout)	(None, 20, 256)	0
lstm_1 (LSTM)	(None, 128)	197120
dropout_1 (Dropout)	(None, 128)	0
dense (Dense)	(None, 1)	129
activation (Activation)	(None, 1)	0

```
-----
Total params: 474,753
Trainable params: 474,753
Non-trainable params: 0
-----
```

From the above image, it can clearly see that there are 474,753 parameters which require training. This is a huge number. By hyper parameter tuning, we will reduce these numbers by grid search method.

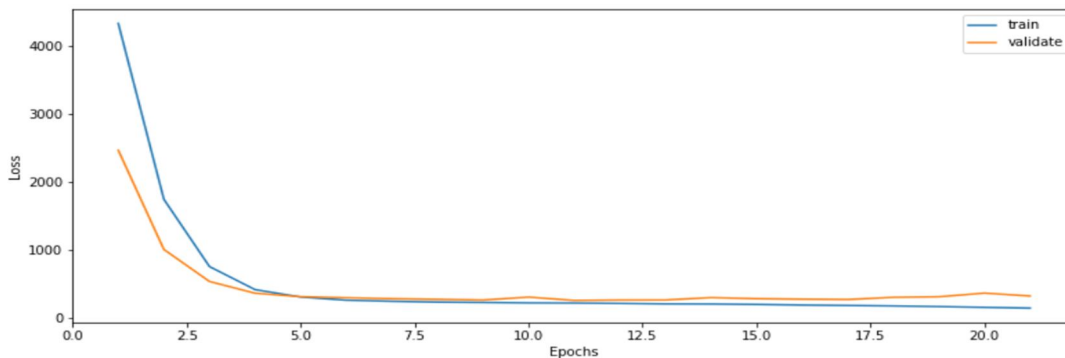


Figure 5: Validation loss vs number of epochs of baseline model for the LSTM for FD00002 data set

Following image shows the predicted vs actual values of RUL values of test and predicted engine data.

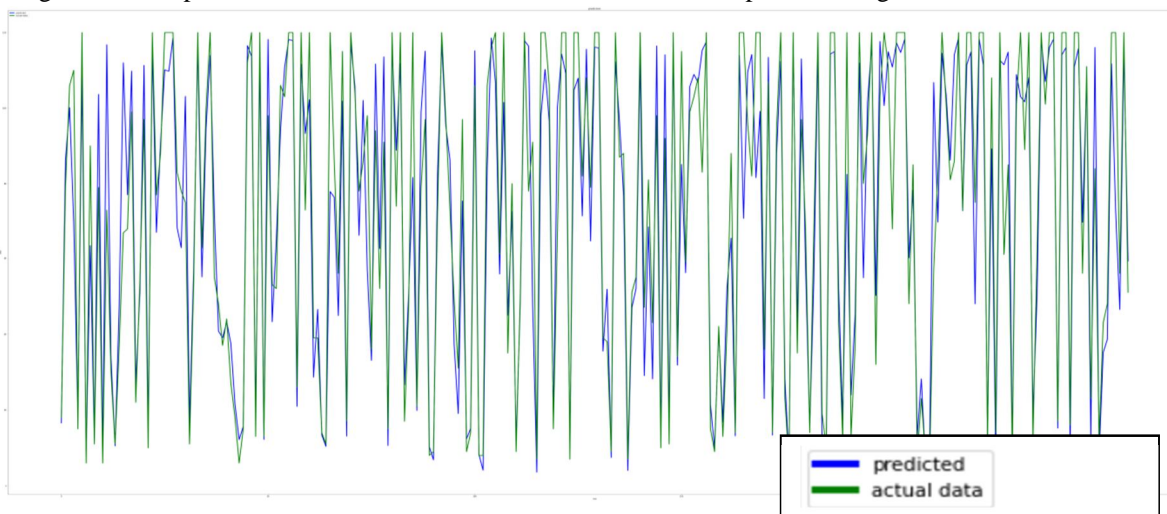


Figure 6: Predicted vs actual RUL prediction of baseline model for the LSTM for FD00002 data base

The above Figure shows that the predicted values and actual test value data are in good agreement and baseline model performed reasonably well.

Following Table shows the performance metrics of baseline LSTM model.

Table 3: Performance metrics of Baseline LSTM model

	RSME	R2 Score
Train Set	12.44	0.90
Test Set	15.04	0.86

From the above table, it can be observed that the R2 score for train set is more than test set. This is a typical case of over fitting and may not be very accurate for the predictions on the unseen data set (test data). So hyper parameter tuning is required to deal with above issue.

V. HYPER PARAMETER TUNING- LSTM RESULTS

A. LSTM Hyper parameters Tuning

Following parameters were considered for the hyper parameter tunings for the LSTM models using random grid search method. Their range is also shown in the below table:

Table 4: Various parameter range used for Hyper Parameter tuning of LSTM Model

Sr. No.	Hyper Parameters	Values
1	first_additional_layer	True, False
2	second_additional_layer,	True, False
3	third_additional_layer	True, False
4	n_neurons	[16, 32, 64, 128]
5	n_batch_size	[8, 16, 32, 64, 128]
6	dropout	[0.1, 0.2]

Following image gives best combination after hyper parameter tuning:

```

Best Combination:
first_additional_layer = False
second_additional_layer = False
third_additional_layer = False
n_neurons = 16
n_batch_size = 8
dropout = 0.2
*****
    
```

Following image shows the results of final LSTM model after hyper parameter tuning.

```

Model: "sequential_44"
-----
Layer (type)                Output Shape              Param #
-----
lstm_88 (LSTM)              (None, 20, 16)          1984
dropout_88 (Dropout)        (None, 20, 16)           0
lstm_89 (LSTM)              (None, 16)              2112
dropout_89 (Dropout)        (None, 16)               0
dense_44 (Dense)            (None, 1)                17
-----
Total params: 4,113
Trainable params: 4,113
Non-trainable params: 0
..
    
```

From the above data, it can be clearly seen that there are the number of trainable parameters drastically reduced from 474, 753 (baseline) to 4, 113. This reduced the overall CPU efforts required without compromise on the accuracy of the prediction model.

Following Table shows the performance metrics of final LSTM model after hyper parameter tuning.

Table 5: Performance metrics of HPT- LSTM model

Sr. No.	RSME	R2 Score
Train Set	14.76	0.865
Test Set	15.12	0.868

From the above table, it can be observed that the R2 score for training and test set is close to 0.87 and almost equal and very decent as per industry norms. This is surely not a case of over fitting and hence this model can be accepted as a final model.

Following image show the predicted vs actual values of RUL values of test and predicted engine data.

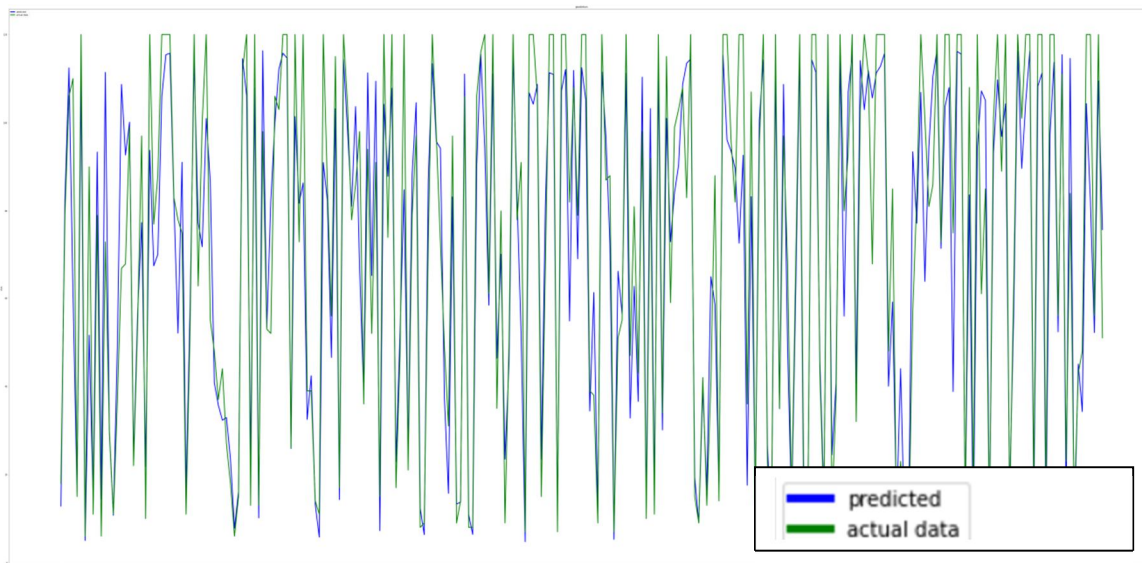


Figure 7: Predicted vs actual RUL prediction of final hyper tuned parameters for the LSTM for FD00002 data base

The above Figure shows that the predicted values and actual test value data are in good agreement and the hyper parameter tuned models performed very well.

VI. RESULTS AND DISCUSSION

In the earlier work by Koen [4], the RUL predictions of FD002 data set were performed using simple Linear Regression and Lagged MLP models. The comparison of this study is presented for the current approach of Hyper parameter tuned LSTM model in the below Table. For each of the models, the results are presented for train and test sets.

Table 6: Performance comparison of HPT- LSTM model with reference work

Model Type	Sr. No.	RSME	R2 Score
Linear Regression Model [4]	Train Set	21.94	0.723
	Test Set	12.82	0.632
Lagged Multi- Layer Perceptron (MLP)Model [4]	Train Set	14.76	0.906
	Test Set	25.35	0.778
Hyper Tuned LSTM model	Train Set	14.76	0.865
	Test Set	15.12	0.868

From this study, following observations can be made:

- 1) The baseline model which is simple Linear regression model performed not so great due to non-linear relation between sensor and RUL. In this case, there is clearly a possibility of further improvement,
- 2) For MLP, the final model is a typical case of over fitting where the R2 score and RSME is more for training data as compared to test data and there is a possibility of further improvement.
- 3) The Hyper-parameter tuned LSTM model performed best with decent R2 score of 0.86 for both training and test dataset.

VII. CONCLUSIONS AND FUTURE PLAN:

In this study, the most robust model for the accurate predictions of Remaining Useful Life is developed. The performance of this developed machine learning model is compared with the existing predictive models based on Simple regression and Lagged MLP techniques. For this purpose a well-known dataset CMAPSS is used for the estimation of correct RUL for test engine data.

Detailed Exploratory Data Analysis (EDA) is performed on raw sensor data to understand the patterns. This helped in removing the sensors having constant values and no patterns in their data.

From the above study, it is evident that the number of trainable parameters of hyper-tuned LSTM drastically reduced from 474, 753 (baseline) to 4, 113. This reduced the overall CPU efforts required without compromise on the accuracy of the prediction model.

Comparative study is of the hyper parameter fine-tuned model made is with simple linear regression and lagged MLP models. The performance of LSTM model is found to be best for both training and test data sets with high level of accuracy. The current approach also eliminated issue of over fitting which was present in the reference models.

The next step of this study is to try other prediction models to improve the efficiency and accuracy of the models.

REFERENCES

- [1] R. C. Staudemeyer and E. R. Morris, (2019) "Understanding lstm—a tutorial into long short-term memory recurrent neural networks," arXiv preprint arXiv:1909.09586, 2019.
- [2] Duarte Pasa, G., Paixão de Medeiros, I., & Yoneyama, T. (2019). Operating Condition-Invariant Neural Network-based Prognostics Methods applied on Turbofan Aircraft Engines. Annual Conference of the PHM
- [3] Koen Peters, (2020), LSTM for predictive maintenance of turbofan engines, Towards Data Science
- [4] Koen Peters, (2020), Lagged MLP for predictive maintenance of turbofan engines. | Towards Data Science
- [5] A. Saxena, K. Goebel, D. Simon, and N. Eklund, (2008), "Damage propagation modeling for aircraft engine run-to-failure simulation," in 2008 International Conference on Prognostics and Health Management, 2008, pp. 1–9.
- [6] Thombre, Ritu & Dhabu, Meera & Gajbhiye, Sanket. (2021), "Remaining Useful Life Estimation in Prognostics and Health Management using LSTM Neural Network and Vector Auto-Regression Models", Proceedings of ACM/ CSI/ EEECS Research & Industry Symposium on IoT Cloud For Societal Applications (IoTCloud'21
- [7] https://keras.io/getting_started/faq/#how-can-i-obtain-reproducible-results-using-keras-during-development
- [8] <https://stackoverflow.com/questions/32419510/how-to-get-reproducible-results-in-keras/59076062#59076062>
- [9] <https://www.onlinemathlearning.com/probability-without-replacement.html>
- [10] The importance of problem framing for supervised predictive maintenance solutions
- [11] <https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.ewm.html>
- [12] https://www.tensorflow.org/tutorials/keras/overfit_and_underfit#add_dropout
- [13] https://www.tensorflow.org/api_docs/python/tf/keras/Sequential



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)