Remaining Useful Life (RUL) Prediction of Mechanical Bearings Using Convolution Neural Network (CNN) and Long Short Term Memory (LSTM)

Bhumireddy Naga Sai Abhijit
Undergraduate Student
Department of CSE
SRMIST
Kattankulathur, Chennai, India
Email: bb3108@srmist.edu.in

Levaku Tharun Sai Reddy
Undergraduate Student
Department of CSE
SRMIST
Kattankulathur, Chennai, India
Email: ll3218@srmist.edu.in

S. Sharanya
Assistant Professor
Department of CSE
SRMIST
Kattankulathur, Chennai, India
Email: sharanys1@srmist.edu.in

***Abstract*—With the availability of data, machine learning models and deep learning models can be trained to predict the likelihood of a certain outcome in a phenomenon. For example, how likely would a user be interested in watching a certain movie given the record of his view history (Netflix recommendation system). The article tackles the problem of predicting the Remaining Useful Life(RUL) of Mechanical Element. As the name suggests, the developed model would predict how long the machine can operate properly without breaking down. Given a record of machine readings over the course of its lifetime, machine learning models can be trained to predict this metric(RUL). RUL can be very helpful in Predictive Maintenance procedures in order to keep the machine reliable and efficient and to reduce machine downtime and operational costs. RUL can also increase machine lifetime. Estimation of RUL depends on the Health Indicator(HI) in Prognostics Data-Driven techniques. Data-Driven approaches calculate the HI from sensor data like vibration signals and with machine learning methods and deep learning methods. With monitoring of sensor data, the Health Indicator can identify the condition of the machine. Basically, HI differentiates the healthy operation and a faulty operation of a machine. In this article, PRONOSTIA Dataset for Bearings(PHM IEEE 2012 Data Challenge Dataset) is used. For every 10 seconds, vibration signals are recorded for 0.1 seconds in the dataset. The recorded signals are 1D raw vibration signals(Time-Domain). The 1D signals are difficult to analyze, so 1D vibration signals are transformed into the 2D image-like features(Time and Frequency-Domain) using CWT(Continuous Wavelet Transform). And CNN(Convolution Neural Network) + LSTM(Long Short-Term Memory) is applied to the acquired features from CWT to calculate the HI. The calculated HI is then used for RUL Prediction.**

***Index Terms*—Machine Learning, Deep Learning, Remaining Useful Life Prediction, Fault Probability Estimation, Mechanical Bearings, Convolution Neural Network Encoder, Long Short Term Memory Unit, CNN + LSTM**

I. Introduction

Every day, we depend on a diverse set of machines. However, each and every machine will break down over time, unless it is maintained. A large number of industrial sectors like automobile, manufacturing, aerospace, etc. are following different maintenance approaches to improve machine reliability, safety, accessibility, and operational quality, which also minimize machine's unplanned downtime and operational costs. One method is to perform reactive maintenance (or) breakdown maintenance. In reactive maintenance, the machine is used to its breaking point and repairs are done only after the machine fails. Reactive maintenance is a standard maintenance approach wherein machines keep operating till the final failure causes them to breakdown. Though this method provides more operating time for a machine before the breakdown, it only gives an inactive response when failure happens, which may have no use and might cause serious damages or accidents. In certain situations where a large number of machines are available as a choice, even a machine has failed and that particular machine's failure is not adverse. This reactive maintenance approach is appropriate. But for certain machines with high-priced parts, one cannot actually endanger running it to failure, as it will be expensive to repair highly damaged parts. Even though, more importantly, it's a safety concern. Because of this, numerous organizations try to prevent failure before it happens by performing regular tests on the equipment. This method is called preventive maintenance (or) scheduled maintenance (or) planned maintenance. In the preventive maintenance approach, the machine is periodically maintained at planned time intervals. Although this improves the efficiency, performance, safety, and reliability of the machine, this also increases the maintenance costs as the machine is checked for even when it is not needed to. A big challenge with preventive maintenance is finding out when to do maintenance. But, if one can predict when machine failure will occur, one can schedule maintenance just before it. This approach is called predictive maintenance[41]. Predictive maintenance predicts the period of time for a machine failure. It also identifies faults in the machine and recognizes which parts need repair. In such a way, one can reduce machine downtime and can increase machine lifetime. One of the predictive maintenance objectives is the prediction of Remaining Useful Life (RUL).

Accurate Remaining Useful Life (RUL) Prediction can be a good predictor of machine health conditions. To observe machine's health conditions, vibration signals are commonly used. As a result, Remaining Useful Life prediction utilizing vibration signals is currently being researched. The machine's Remaining Useful Life (RUL) can be predicted with any of the two methods : (1) model-based method, (2) data-driven method. Model-based methods make use of physics-related models to predict how quickly a machine may fail. Model-based approaches are hard to implement because the models of these methods are complex to understand and building reliable physics-related models is quite challenging. Thanks to their benefits over model-based approaches, the concept of data-driven methods for Remaining Useful Life (RUL) prediction received considerable recognition lately. Data-driven approaches create prediction models or machine learning models. Prediction models take data as an input, apply prediction on the input data, and then return an output based on that predicted input data. In data-driven techniques, the premier step is to collect data under different operating conditions, to build a machine learning model. We collected PRONOSTIA Bearing Dataset (PHM IEEE 2012 Data Challenge Dataset). If the data fetched is not clear or not enough, data-driven approaches may be constrained. Deterioration i.e. the damage possibility of a machine is calculated easily and accurately when large amounts of that machine's data are available, and with data-driven approaches, the damage process of a machine is calculated even without knowing much about the machine. Therefore, compared to model-based approaches, data-driven approaches are simple to implement.

The remaining useful life (RUL) prediction depends on health indicators (HI) in data-driven methods. The remaining useful life prediction output is derived from the health indicator. Well-constructed health indicators with good facilities can accurately inspect, check and keep track of mechanical elements breakdown. For that reason, the extraction of ideal features from sensor data viz. Vibration signals (horizontal and vertical acceleration vibration signals) become the main and important functions. The features should be thoroughly investigated before extracting data from them. Most data analysis can be performed in two ways - time-domain analysis and frequency-domain analysis, as well as they derive vibrational signal features. In time-zone analysis, vibration signals can be examined and determined based on time. Frequency domain analysis monitors vibration signals based on frequency. With time-field analysis, the variation of vibration signals over time is visualized. On the other hand, frequency-domain analysis visualizes what not. Of the vibration signals present at a specific period. Features derived from time-domain analysis perform well and are accurate for stationary signals, although these features respond quickly to random variations within vibration signals and achieve randomness, these time-domain features allow for vibration signals She gives. Are the same. Unlike time-zone analysis features, features of frequency-domain analysis can correctly interpret vibration signals because they determine more insight about signals than time-zone analysis features. Characteristics of frequency-domain analysis can assess and separate frequency parts, which helps to understand more insights than time-domain analysis. Mechanical bearing vibration signals are generally variable and at the same time these signals have undetermined faults in high noise environments. Therefore time-frequency domain analysis is an appropriate method to examine mechanical bearing vibration signals because the vibration signals of mechanical bearing are interconnected. Features derived from time-frequency domain analysis have properties of both time components and frequency components.

In this paper, we predict the Remaining Useful Life (RUL) of Mechanical Bearings from Health Indicators (HIs) by applying Continuous Wavelet Transform (CWT), Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM) network.

The remaining part of the paper is set out in the following way. Section II reviews the related work. Section III presents the proposed work. Section IV is the implementation and results of the developed work. Finally, in Section V, we conclude with a discussion and mention future research topics.

II. Related Work

During the course of a review of present systems on the prediction of Remaining Useful Life (RUL), we came across different RUL prediction models. And, we noticed that in this domain, huge amounts of work are being done to generate a prediction model that performs well on estimating the Remaining Useful Life (RUL) of mechanical elements because the RUL Prediction helps in decreasing machine's unplanned downtime and operational costs, increasing efficiency of the machine, improving the machine reliability and accessibility. Below we are mentioning some of the works done related to our work.

Hian Leng Chan and others[2] presented a Multiple Recurrent Neural Networks framework and extracted Time-Frequency features using Wavelet Transform and Empirical Mode Decomposition for the estimation of Remaining Useful Life (RUL). The researchers also examined the impact on processing time and prediction accuracy of Multiple Recurrent Neural Networks and Time-Frequency features and they found that the proposed Method shows better prediction accuracy with Empirical Mode Decomposition compared to Wavelet Transform and other traditional Remaining Useful Life approaches, but processing time is less with Wavelet Transform compared to Empirical Mode Decomposition.

Boyuan Yang and others[5] proposed a double-Convolutional Neural Network (double-CNN) architecture for predicting the Remaining Useful Life (RUL). This article makes some significant advancements for enhancing the efficiency of RUL prediction. The architecture is immediately operated by primitive vibration signals to fix the feature extraction issues and to maintain the appropriate information which is helpful for prediction. To construct a Remaining Useful Life prediction model, in this work, the researchers calculated slow and sudden increasing trends. To resolve the issue of Remaining Useful Life time and variation in results, this paper implements a mediating variable.

Brigitte Chebel-Morello and others[7] predicted the machine's Remaining Useful Life from sensor values without an intermediary i.e. without calculating the breakdown threshold and deterioration condition of a machine, and SVR model is utilized to form the explicit interaction among health indicators and sensor values so that the prediction of Remaining Useful Life is done immediately after the degeneration. Finally, implemented a wrapper variable selection approach prior to the model training in order to improve the model efficiency.

Liang Gao and others [12] introduced a data-driven method that uses traditional neural networks (CNN) to identify defects and a transformation technique that transforms signals into 2D images. CNN is by default an expert model for correctly extracting features from unprepared data. Feature extraction is quite important in data-driven methods and regression problems because it is helpful in understanding more insights about data and it also highly affects the end result, but feature extraction techniques are complex processes. So to avoid complexity in the work, the researchers used the CNN because it has the ability to extract the feature directly from the input image. The authors used 3 different data sets to evaluate the neural network model: (1) self-priming centrifugal pump fault diagnosis data set, (2) axial piston hydraulic pump fault diagnosis data set, (3) motor. Bearing Data Set. The suggested approach achieves high efficiency over all 3 data sets.

Abdenour Soualhi and others [13] used a method that integrates one technique and two machine learning algorithms. The 2 algorithms implemented in the paper are support vector machine (SVM) and support vector regression (SVR) and the technique used to estimate ball bearings is the Hilbert Huang transform (HHT). This work used the Hilbert Huang transform to derive HI from bearing vibration signals that help monitor the bearing degradation process. The support vector machine is used to study the derived HI to identify faults. And support vector regression is applied for the remaining useful life prediction.

Jianjing Zhang and others[21] applied LSTM (Long Short-Term Memory) network to estimate the Remaining Useful Life (RUL). Employed NASA's C-MAPSS data set to examine the proposed approach. This paper particularly used LSTM to explore deviations and irregularities present in time-series which helps in monitoring the machine decaying process. For evaluating efficiency of a machine and estimating machine's Remaining Useful Life, the authors wielded on board sensor data. Sensor data is used to generate a health index (HI) which describes performance of a machine. After that, bi-directional Long Short-Term Memory network is applied to monitor the HI deviation. Estimation of Remaining Useful Life is done with respect to HI variability. Bi-Directional LSTM made the monitoring and prediction process less complex.

Kwok Leung Tsui and Nan Chen[24] performed condition based monitoring and RUL Estimation of bearings with a 2-phase deterioration model. This work combined deterioration data of bearings previous performance data alongside undamaged data by applying Bayesian architecture. The Bayesian architecture improved the reliability in predicting the damaged features of the bearings. The authors calculated relationship between various insights from predictions clearly which enhanced the Remaining Useful Life Prediction efficiency.

P.S. Heyns and S.A. Aye[33] estimated bearings RUL by making use of an adaptive GPR (Gaussian Process Regression) algorithm. In this article, several mean and co-variance methods are used to evaluate nine Gaussian Process Regression frameworks for all the 3 bearings under 3 various working conditions which resulting in a twenty seven Gaussian Process Regression frameworks overall for each bearing. Using co-variance methods, the constant, linear, and zero mean models constructed. The researchers extracted 3 co-variance models and 2 mean models which are efficient with the help of MAPE(Mean Absolute Percentage Error) and RMSE(Root Mean Square Error) among twenty seven GPR frameworks. The extracted models are then linked together and built an integrated Gaussian Process Regression model which predicted the RUL of bearings more accurately compared to other basic Gaussian Process Regression models.

Feng Yang and others[36] predicted the electrical machines Remaining Useful Life using HI Predictive technique. The developed HI Predictive approach predicted Remaining Useful Life (RUL) as follows: estimated HI (Health Index) through input data, after that the calculated HI was then mapped to Remaining Useful Life (RUL). An unchanging, constant-based Health Index Smoothing methodology was implemented to increase the predictability of Health Index. The separation of Time Domain features, Frequency Domain Features, Time-Frequency Domain features were carried out and neural network is applied to analyze the extracted features and to estimate the Health Index. Evaluated the final Remaining Useful Life by enforcing an ensemble approach which concatenated various network predictions.

Chau Yuen and others[35] developed a network model for factory-made equipment's Remaining Useful Life Prediction. The constructed framework is an ensemble of Long Short Term Memory network plus layers of Temporal Convolution along Data Augmentation technique. In this research work, temporal convolutions network is designed with short term dependencies and long term dependencies altogether to improve the network performance. Temporal Convolution network made up of convolution layers, max pooling layers, and activation function. The authors implemented the Long Short Term Memory network in the framework to optimally manage the long term dependencies . Data standardization technique is applied for learning data and testing data for faster training as this technique adjusts the data to a specific range. Data augmentation is applied to the learning data and then inputted the learning data to the framework. The developed framework estimated the Remaining Useful Life by acquiring a knowledge of hidden time changes and taking out important features from input sensor values.

A. *Inference from the Related Work*

By reviewing some of the research works done associated to our work, Remaining Useful Life (RUL) prediction, we learned that RUL prediction is a regression problem. Since RUL Prediction is a regression problem, the RUL prediction of mechanical element (mechanical bearing) can be calculated using that mechanical bearing's previous performance data.

III. Proposed Work

This research article focuses on predicting the Remaining Useful Life (RUL) of Mechanical Bearings. We developed a data-driven methodology that incorporates Convolution Neural Network (CNN), Continuous Wavelet Transform (CWT), and Long Short-Term Memory (LSTM) Network for Remaining Useful Life (RUL) prediction.

A. *Modules of the proposed work*

A.1 Data Collection:

To create a prediction model (Machine Learning model or Deep Learning model). The origin is the collection of data of the specific machine or mechanical element under diverse working conditions which we are using for prediction. For our work, we collected PRONOSTIA Bearing Data set (PHM IEEE 2012 Data Challenge Data set).

The collected data set is comprised of 6 bearings training or learning data sets: Bearing1_1, Bearing1_2, Bearing2_1, Bearing2_2, Bearing3_1, Bearing3_2 and 11 bearings test data sets: Bearing1_3, Bearing1_4, Bearing1_5, Bearing1_6, Bearing1_7, Bearing2_3, Bearing2_4, Bearing2_5, Bearing2_6, Bearing2_7, Bearing3_3. We used learning or training data sets to construct a prediction model and utilized test data sets to check the constructed prediction model's performance. Each data set contains several number of data files. Every data file in the data sets(learning and test) contain the information of horizontal acceleration vibration signals and vertical acceleration vibration signals. The sampling rate of the vertical acceleration and horizontal acceleration vibration signals is 25.6 KHz. The sampling rate specifies how many samples i.e. vibration signals are collected per one second. Sampling rate also called as sampling frequency, therefore 25600 vibration signal samples of both horizontal acceleration and vertical acceleration are acquired per one second. In the data files, one vibration sample or vibration data point recording of 0.1 seconds is collected for every 10 seconds. Hence, each data file consists of 2560 data points (horizontal acceleration vibration and vertical acceleration vibration data points)

Fig. 1 shows the horizontal and vertical acceleration vibration signals for Bearing1_3 dataset

A.2 Data Preprocessing(or Data Preparation) and Signal Processing:

Preprocessed data is needed for building and training any machine learning model and deep learning model. The data

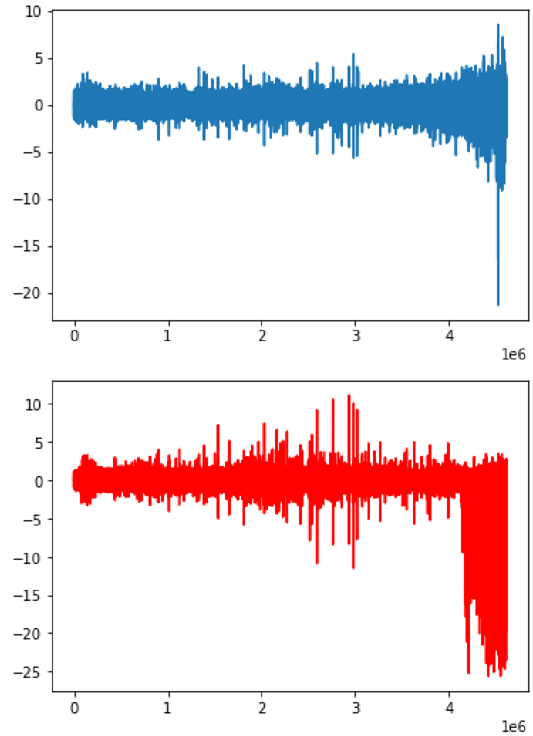


Fig. 1: 1d horizontal (top plot) and vertical (bottom plot) vibration signals

we collected is the raw data, but machine learning models and deep learning models are unable to comprehend the raw data. Henceforth, we applied data preprocessing to the collected raw data set. The raw data were cleansed and formatted by applying data preprocessing, to made the raw data ideal for machine learning and deep learning models construction and training. Data Preprocessing is a vast process comprises of several sub-processes. We employed Data Normalization technique after applying CWT to the 1D collected data to normalize the coefficients of 2D CWT data. To achieve Data Normalization we performed Data Re-scaling method A.K.A min max normalization to re-scale or adjust all the 2D data to a particular range (say 0 to 1). (1) here represents min max normalization.

$$coef = \frac{coef - coef.min()}{coef.max() - coef.min()} \tag{1}$$

Signal Processing visualize, evaluate, and control vibration signals. In our work, we extracted 2D features i.e. time-frequency domain features of horizontal acceleration vibration signals and vertical acceleration vibration signals from 2D CWT image data by implementing signal processing. The mechanical bearing's deterioration can be easily visible with the help of time-frequency domain features because time frequency features contains more information about the vibration signals, in regression problems like Remaining

Useful Life Prediction, more data means fast and easy analysis.

A.3 Training the Model

Any Machine Learning Model and Deep Learning Model should undergo training to perform prediction operation. In Data Preprocessing, we split the collected learning / training dataset of PRONOSTIA Bearing dataset into Validation data and train data. First, we preprocess the train data and using that preprocessed train data, drawn out important features, and implemented algorithms train the model to predict.

A.4 Validating the Model

Using validation data, the model validation can be performed. During data preprocessing, we divide the learning dataset into train data + validation data. Validation data is the set of data that is separated from learning dataset to validate or check the trained model. We divided 10 percent data from learning data and considered that data as validation data, whereas the remaining 90 percent data is train data. Validation data is used to validate the model whether the model is working correctly or not on the data which the model is not trained. One of the major reasons we need validation data is to make sure that our model is not over- fitting to the data in the learning dataset.

When the model performs too well and make accurate predictions on the train data, and if the same model does not performs well and does not make accurate predictions on the data that model wasn't trained (for example: validation data, test data), then that model is said to have over-fitting. So, to avoid over-fitting, during training, validate the model on our validation data and check whether the results the model giving for the validation data is just as good as the results the model giving for the train data. So that we know whether our model is over-fitting or not.

A.5 Testing the Model

Model testing can be performed using test data. Test data is the set of data used to test the model in terms of prediction accuracy and performance after the model has been already trained and validated.

Fig. 2 below represents the functional pattern of modules.

B. Methodology of the proposed work

In our work, we implemented two deep learning algorithms, Convolution Neural Network (CNN), Long Short Term Memory (LSTM) Network along with a machine learning technique Continuous Wavelet Transform (CWT) to estimate the fault probability or Health Indicator (HI) and to solve the Remaining Useful Life (RUL) Prediction problem.

B.1 CWT (Continuous Wavelet Transform):

The purpose of Continuous Wavelet Transform (CWT) is to generate a time-frequency depiction of a vibration signal which delivers an accurate positioning of both time plus frequency of the signal. It is also used for computing the vibration signal varying features. The signals are represented as wavelets in Wavelet Transform. The CWT transforms the signals into wavelets. Wavelets are wave shaped vibrations of magnitude which initiates at zero, progresses, later reduces again to zero. The wavelets produced from CWT provides well-defined and comprehensible interpretation of time and frequency components which helps in understanding the bearings deterioration process clearly. The Continuous Wavelet Transform contains several in-built CWT wavelets like morlet wavelet, gaussian derivative wavelet, frequency b-spline Wavelet, mexican hat wavelet, shannon wavelet, complex morlet wavelet, etc. which can be accessed using python programming. And among them, we used morlet wavelet python implementation i.e. CWT Morlet PyWavelet as it shows better outcomes for regression problems and compared to other wavelet types and fourier transform. Theoretically, the Morlet Wavelet works well for time-frequency vibration signals. CWT of a signal is calculated as follows [3].

$$wt(a,b) = |a|^{-\frac{1}{2}} \int_{-\infty}^{\infty} x(t)\Psi^*(\frac{t-b}{a})dt \tag{2}$$

where Ψ^* is Morlet Wavelet [1]

$$\Psi^*(t) = exp^{\frac{t^2}{2}} \cos(5t) \tag{3}$$

In this work, the CWT Morlet PyWavelet is applied to reform the recorded 1D vibration signals of PRONOSTIA Bearing Learning Dataset into 2D CWT image like features or time-frequency domain image features as shown in Fig. 3. In Fig. 3(a) and in Fig. 3(b), x-axis represents time and y-axis represents frequency. 1D signals contain only time field intelligence. It is quite tough to interpret the signals in only one dimension(time), so converted the signals from one dimension to two dimensions. 2D signals i.e. 2D Continuous Wavelet Transform image features carries the information about the signals from both time and frequency domains which helps in visualizing the damage process of bearings perfectly.

B.2 Convolution Neural Network (CNN) Encoder:

Convolution Neural Network (CNN) is employed to evaluate the transformed feature images. CNN can appropriately assess the images, obtain more useful information from an input image. The image inputs of CNN are arrays of pixel values (px). CNN takes input as [NxCxHxW] shape: where N is the batch size, C is the no. of channels or no. of filters, H is the height of an image in pixels, W is the width of an image in pixels. Convolution Neural Network works effectively with image like data.

As demonstrated in Fig. 4, we inputted 2D CWT images of horizontal and vertical acceleration of size 2x128x128 to the CNN encoder. CNN Encoder converts the image input data into 1D linear continuous vector. The loaded image first undergoes convolution operation with 3x3 filters. Each pixel value inside the image is multiplied one-by-one with filters in the convolution process. In CNN, the main objective of convolution is to extract necessary features from the input image. Filters also called as kernels or channels or

[1] https://pywavelets.readthedocs.io/en/latest/ref/cwt.html

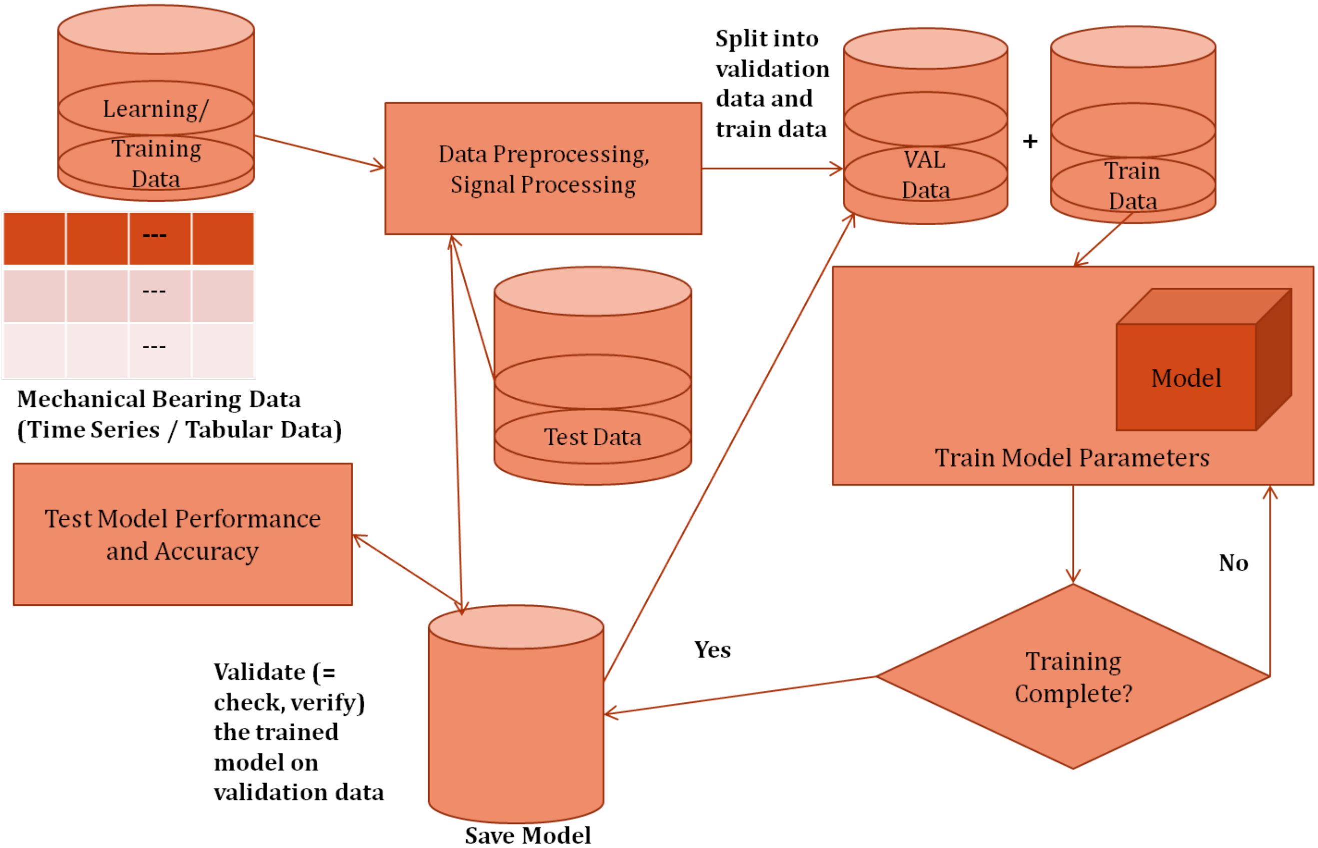
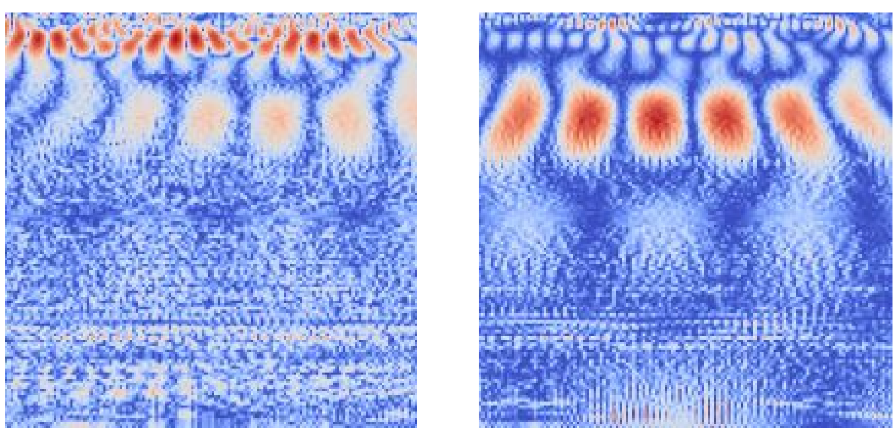


Fig. 2: **Modules flow diagram**



(a) Horizontal acceleration features image (b) Vertical acceleration features image

Fig. 3: Converted 2D CWT feature images

feature detectors used to evaluate the pixels of an image input. Filter is represented as matrix. The values in the filter are considered as weights. At first, Values in the filter are randomly specified when constructing the CNN, which then repeatedly update the values in the filter as the CNN is trained. In CNNs, filters are not defined. The value of each filter is learned during the training process. By being able to learn the values of different filters, CNNs can find more meaning from images that humans and human designed filters might not be able to find. This filter matrix moves over the complete input image from top left to bottom right and performs matrix multiplication between the weights of the filter matrix and pixel values of the input image. After engaging convolution operation, applied stride of 1 to the input image. Stride is the no. of pixel values shifts over the input image matrix and those shifts are done only from left to right. And then applied padding of 1. padding means simply a process of adding layers (rows and columns) of zeros around an input image. Advantage of padding: padding prevents the image from shrinking, it also helps preventing dimensionality reduction. Batch normalization is also applied. Batch refers to the collection of data, instead of inputting each data sample at a time we divided the dataset into batches of data and inputted each batch at once, and each batch contains 64 data samples. Batch normalization means re-scaling all the data i.e. pixel values of a batch to a particular range (say [-1,1]). Finally applied ReLU (Rectified Linear Units) activation function, that will output the positive values or non-negative values as it is without change, whereas output the negative values as zero (0). Advantage of ReLU: ReLU allows the models to learn faster and perform better, avoids overfitting. After applying convolution, stride, padding, batch normalization and ReLU, we obtained a convolved image feature of 16x128x128 proportions. We then applied maxpooling with 2x2 filters and stride of 2 to the convolved image to get a pooled feature image of 16x64x64. We applied convolution + stride + padding + batch normalization + ReLU + maxpooling until the image shape becomes 128x8x8. After that we flattened 128x8x8 image to a 1D linear continuous vector of 8192 size and passed this 8192 size vector to fully connected layers or linear layers. The fully connected layer applies linear transformation, dropout technique and ReLU activation function to the input flattened vector which reduces the size of the vector for inputting that vector into LSTM.

B.3 Long Short Term Memory (LSTM) unit:

There is inherent time dependence in the input data of mechanical bearing for RUL prediction. Using an LSTM network is an appopriate technique to learn patterns in a time-series data. Time-series data is a set of observations i.e. data usually collected at discrete and equally spaced time intervals. Here, in our collected dataset, vibration signals are collected at every 10 seconds that is different time and equal time intervals. Hence our data is time-series data. And also LSTM is a sequence model which performs sequence modeling.

Sequence modeling is the process of generating a sequence of values by analyzing a series of input values. These input

values could be time series data. Basically Sequence Modeling is a task of predicting what comes next. In sequence modeling, the current output is dependent on the previous inputs and the length of the inputs is not fixed. Data of shape [NxLxCxHxW] for all bearing data files doesn't fit in memory of LSTM, so they need to be packed into sequences at run time. Here N, L, C, H, W represents batch size, sequence length, number of channels, height, width respectively. There are certain problems with sequence modeling: can't model long-term dependencies, don't preserve order of data points, no parameter sharing. To avoid these problems Long Short Term Memory is used, because LSTM RNN: deal with variable length sequences, maintain sequence order, keep track of long-term dependencies, share parameters across the sequence.

B.4 CNN + LSTM Model:

Input data is still represented as 2D CWT images i.e. time-frequency CWT feature images, but a window of pre-defined frames (say 5) is used as a sequence. First passed through a CNN architecture for encoding and then encoded feature vector sequence is passed through LSTM architecture, hence CNN + LSTM architecture.

In our CNN + LSTM model, Fig. 6, first we inputted 2D CWT images to the CNN for encoding. CNN Encoder converts the 2D image input data into 1D linear continuous vector by flattening. The transformed feature vector sequence is then inputted to LSTM units. As pictured in Fig. 6, LSTM units analyze the inputted vector sequences at different time steps: t, t+1, t+2, t+3, t+4 and output the final time step : t+4 LSTM unit vector sequence, from which we calculate the HI(Health Indicator) or fault probability or failure probability.

- LSTM unit at time step t takes hidden layers or hidden states and encoded vector sequence of CNN Encoder of t time step image as input.
- LSTM unit at time step t+1 takes output of LSTM unit at t time step and current encoded vector sequence i.e. encoded vector sequence of CNN Encoder of t+1 time step image as input.
- LSTM unit at time step t+2 takes output of LSTM unit at t+1 time step and current encoded vector sequence i.e. encoded vector sequence of CNN Encoder of t+2 time step image as input.
- LSTM unit at time step t+3 takes output of LSTM unit at t+2 time step and current encoded vector sequence i.e. encoded vector sequence of CNN Encoder of t+3 time step image as input.
- LSTM unit at time step t+4 takes output of LSTM unit at t+3 time step and current encoded vector sequence i.e. encoded vector sequence of CNN Encoder of t+4 time step image as input.

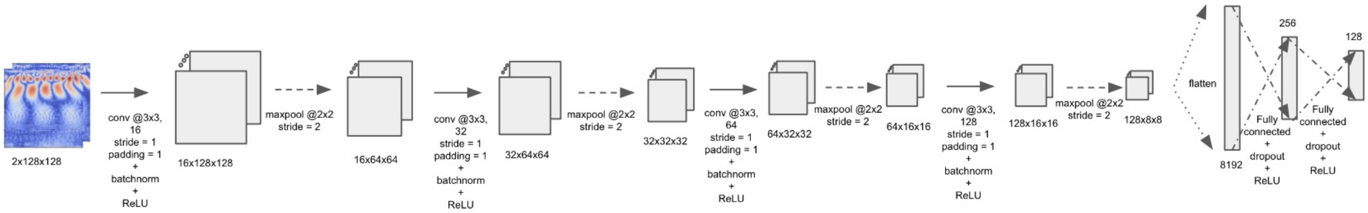


Fig. 4: CNN encoder

IV. Implementation and Results

The entire implementation is performed in Colaboratory platform using python programming. We utilized both CPU - Central Processing Unit and default GPU - Graphics Processing Unit provided by Colaboratory during the application process. Converted Bearing Learning Dataset into pickle files (.pkz) for easy data access and faster data retrieval. These bearing pickle files contains 1D data (horizontal acceleration and vertical acceleration vibration signals). Performed Continuous Wavelet Transform with Morlet Wavelet on 1D vibration signals which returned the 2D CWT feature images. Normalized the obtained feature images to a specific range, say 0 to 1. Applied signal processing to extract the time-frequency features which helps in differentiating the healthy operation and the faulty operation of mechanical bearing as shown in Fig. 10 and Fig. 11. Since our dataset is a run to failure dataset, at starting the functionality of bearing is in good condition and eventually fails while progressing. Splitted the learning dataset into train data and validation data. Train data to train the model, validation data to validate the model. Transformed both validation data and train data into pickle files.

Imported torch Dataset and DataLoader for easy accessing of data. A lot of effort in solving any deep learning problem goes in to preparing the data. torch package provides many tools to make data loading easy: if data loading is easy feeding the data into the model is easy, and to make the code more readable.

Initially we calculated the fault probability with pure CNN model as shown in Fig. 5 (created by adding additional fully connected layer with one output neuron to the CNN encoder from Fig. 4). To train the model, we switched from CPU device to GPU device in colaboratory in order to reduce the training time. We utilized GPU for computation with CUDA tensor. During training, first we calculated the loss function to measure the difference between actual expected value and the model predicted value. Loss function evaluates how well our algorithm performs on our dataset. If our model

predictions are totally off i.e. wrong, our loss function will output a higher number. If our predictions are pretty good, our loss function will output a lower number. Since we adjust our algorithm to try and improve our model, our loss function will tell us whether our model is improving or not, loss function will tell us whether our model predicting correctly or not. Since RUL Prediction is a regression problem, MSE (Mean Squared Error) Loss is used. MSE Loss is estimated as (4). MSE loss is the average of squared differences between actual expected values and predicted values.

$$M = \frac{(a1 - p1)^2 + (a2 - p2)^2 + ... + (aN - pN)^2}{N} \qquad (4)$$

where, M = Mean Squared Error Loss
a1, a2,- - -, aN = actual values or target labels
p1, p2,- - -, pN = predicted values
N = total no. of observations

Loss helps us to understand how much the predicted value differ from actual value. we then used this loss to train our network model such that it performs better. Basically we taken the loss and reduced it, because a lower loss means our model is going to perform better. We applied Adam (Adaptive Moment Estimation) optimizer or optimization algorithm for reducing the loss and to provide the accurate results possible. Implemented learning rate scheduler to improve the optimization process. Learning rate scheduler adjust the learning rate for better outcomes during the training loop based on the no. of epochs. The learning rate controls how much to change the model in response to the estimated error each time the model weights are updated. Epochs are the batches of data samples used to train the artificial neural network. Calculated the loss on train data and validation data for 30 random epochs. TABLE I shows the CNN Model loss results on train data and on validation data, and complete history of losses over all the epochs is illustrated in Fig. 7

TABLE I: CNN Model train loss and validation loss results

Epoch	Train Loss	Val Loss
1 / 30	0.6206	0.0457
5 / 30	0.0153	0.0172
10 / 30	0.0098	0.0186
15 / 30	0.0053	0.0088
20 / 30	0.0048	0.0087
25 / 30	0.0041	0.0085
30 / 30	0.0040	0.0083

Fig. 12 describes the pure CNN model performance in estimating the probability of faults on training dataset. In Fig. 12, the blue dots represent predicted fault probability values of train data, the red dots represent predicted fault probability values of validation data, and the black line represent expected failure probability values. Expected values are the ideal values. For example, if the actual value or true label is 0.1, then ideally the predicted value should also 0.1.

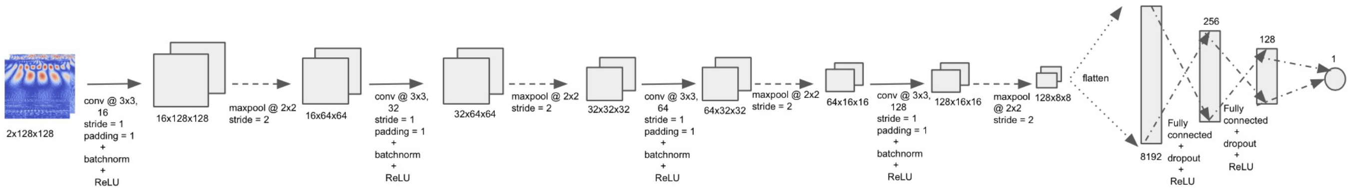


Fig. 5: Exclusive CNN model for Health Indicator (HI)

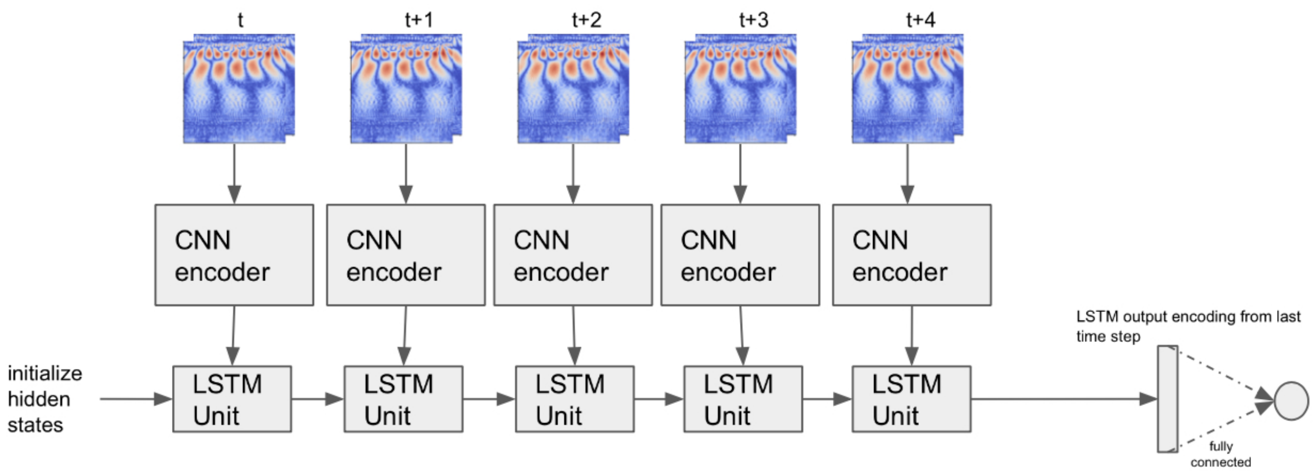


Fig. 6: CNN + LSTM for Health Indicator (HI)

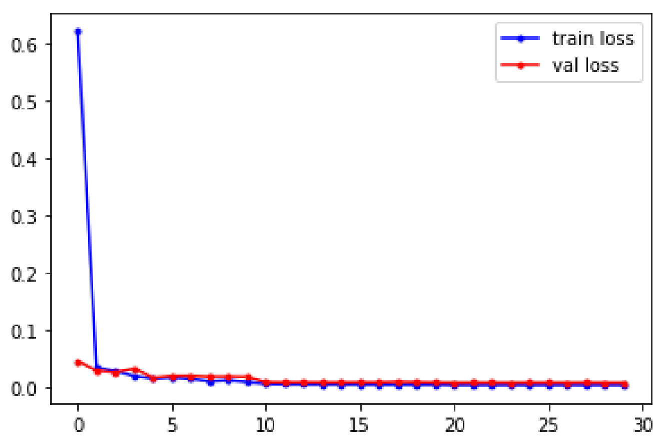


Fig. 7: CNN training loss

Basically expected labels are plotted to visualize the train and val results are how much closely related to the ideal expected results.

For CNN + LSTM Model, we prepared the data, packed the data into sequences of shape [(NxL)xCxHxW]. As depicted in Fig. 6 first passed the 2D CWT images through CNN for encoding and then inputted encoded output to the LSTM for fault probability prediction. TABLE II shows CNN + LSTM model loss results on train and validation data, and full loss history over all training epochs is represented in Fig. 8

TABLE II: CNN + LSTM train loss and validation loss results

Epoch	Train Loss	Val Loss
1 / 30	0.0364	0.0983
5 / 30	0.0120	0.1147
10 / 30	0.0038	0.0834
15 / 30	0.0024	0.0822
20 / 30	0.0014	0.0706
25 / 30	0.0007	0.0618
30 / 30	0.0006	0.0334

Fig. 13 exhibits the fault probability values with CNN + LSTM architecture. The black line is the expected values, the blue dots are train data values, the red dots are validation data values.

We propose to predict the RUL of the system as follows,

- Pass the test data till current time index through the architecture and store the predicted fault probability values
- Then fit the predicted fault probability values and corresponding time indices through a regressor to extrapolate and find the time index when the fault probaility exceeds a predefined threshold to consider occurrence of failure. Here we used Linear regression and Guassian Process Regression (GPR) provided by sklearn python toolbox, and set the fault probability threshold to be 0.9

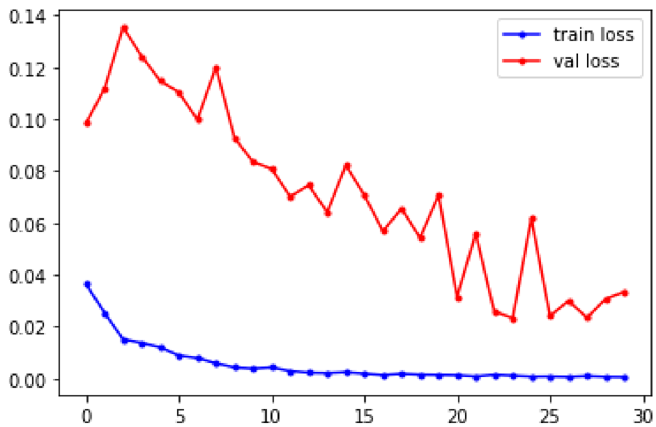


Fig. 8: CNN + LSTM training loss

Test set	Actual RUL
Bearing1_3	5730 s
Bearing1_4	339 s
Bearing1_5	1610 s
Bearing1_6	1460 s
Bearing1_7	7570 s
Bearing2_3	7530 s
Bearing2_4	1390 s
Bearing2_5	3090 s
Bearing2_6	1290 s
Bearing2_7	580 s
Bearing3_3	820 s

Fig. 9: True RUL values of test data

RUL is estimated from the estimated failure occurrence time and current time as follows,

$$RUL = predicted\ fault\ time - current\ time \quad (5)$$

Fig. 14 shows the results of CNN + LSTM architecture on test dataset. Similar results were noticed with CNN architecture as well. Obtained results seem to be inaccurate.

True RUL values for test data as provided by official PHM challenge are showin in Fig. 9. Refer [20].

V. CONCLUSION AND FUTURE WORK

Continuous vibration measurements are used for predicting fault probability and RUL for the bearing setup. Inherently input exhibits characteristics of time series data. Hence this article proposes to explore LSTM combined with a CNN encoder to obtain reliable predictions. However on the training and validation sets, it appears that the just CNN architecture perform better compared to a CNN+LSTM architecture (refer to Fig. 10 and Fig. 11).

Both architectures performed poorly on the test data, showing that the networks failed to generalize well (refer to figure where test data results follow no intuitive pattern i.e. Fig.14). It seems both architectures have over-fitted the training data.

This behavior is possibly due to following reasons,

- Input horizontal and vertical vibration 1-D signals are converted into 2-D feature maps by performing CWT. And during the conversion, to ensure that the feature maps are of size 128×128, windows of size 20 are taken from the 1-D signal and average value is used. This conversion process might not be appropriate and needs to be explored further.
- Both architectures proposed (CNN and CNN+LSTM) may be larger than what is required for the data and hence they overfit the training data
- When focused more on the original 1-D input signals, though the data represent a run-to-failure experiment, the data for the most part until the end of the run seems to be within $\pm 5m/s^2$ and at the end exceeds $20m/s^2$ in magnitude. Which probably means probability of failure cannot be assumed to be linearly changing as it was used for training. When data is with in nominal range it should have low fault probability and that would increase as the values go into the abnormal range. This seems to be the main factor causing the poor performance of both models on test data as training labels might not be appropriate for the learning task.

For future work, we propose the following,

- Improve feature extraction scheme and get more relevant training labels
- Instead of converting to 2-D feature maps, focus more on generating training labels and use 1-D signals directly as input to 1-D Convolutions for encoding combined with LSTM to predict fault probability

REFERENCES

[1] J. Lin, L. Guo, N. Li, T. Yan, and Y. Lei, “Machinery health prognostics: A systematic review from data acquisition to rul prediction”, *Mechanical Systems and Signal Processing*, vol. 104, pp. 799–834, 2018.

[2] Hian Leng Chan, Jihoon Hong, Qiushi Wang, and Xueheng Qiu, “Remaining Useful Life Prediction Using Time-Frequency Feature and Multiple Recurrent Neural Networks”, *24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 916-923, 2019.

[3] Youngji Yoo, and Jun-Geol Baek, “A Novel Image Feature for the Remaining Useful Lifetime Prediction of Bearings Based on Continuous Wavelet Transform and Convolutional Neural Network”, *Applied Sciences*, vol. 8, no. 7, pp. 1102, 2018.

[4] Feng Jia, Jing Lin, Liang Guo, Naipeng Li, and Yaguo Lei, “A recurrent neural network based health indicator for remaining useful life prediction of bearings”, *Neurocomputing*, vol. 240, pp. 98-109, 2017.

[5] Boyuan Yang, Enrico Zio, and Ruonan Liu, “Remaining Useful Life Prediction Based on A Double-Convolutional Neural Network Architecture”, *IEEE Transactions on Industrial Electronics*, vol. 1, no.1, pp. 1-10, 2019.

[6] F. Jia, N. Lu, S. Xing, and Y. Lei, “Deep normalized convolutional neural network for imbalanced fault classification of machinery and its understanding via visualization”, *Mechanical Systems and Signal Processing*, vol. 110, pp. 349-367, 2018.

[7] Brigitte Chebel-Morello, Emna Laajili, Farhat Fnaiech, Noureddine Zerhouni, Racha Khelif, and Simon Malinowski, “Direct Remaining Useful Life Estimation Based on Support Vector Regression” *IEEE Transactions on industrial electronics*, vol. 64, no. 3, pp. 2276-2285, 2016.

[8] Huijun Gao, Okyay Kaynak, Shen Yin, and Xianwei Li, “Data-Based Techniques Focused on Modern Industry: An Overview”, *IEEE Transactions on Industrial Electronics*, vol. 62, no. 1, pp. 657-667, 2015.

[9] Meng Ma, and Zhu Mao, "Deep Convolution-based LSTM Network for Remaining Useful Life Prediction", *IEEE Transactions on Industrial Informatics*, vol. 17, no. 3, pp. 1658-1667, 2020.

[10] Chuang Sun, Meng Ma, and Xuefeng Chan, "Deep coupling autoencoder for fault diagnosis with multimodal sensory data", *IEEE Transactions on Industrial Informatics*, vol. 14, no. 3, pp. 1137-1145, 2018.

[11] Dali Cao, Huibin Sun, Xia Kang, and Zidong Zhao, "A hybrid approach to cutting tool remaining useful life prediction based on the Wiener process", *IEEE Transactions on , Reliability*, vol. 67, no. 3, pp. 1294-1303, 2018.

[12] Liang Gao, Long Wen, Xinyu Li, and Yuyan Zhang, "A New Convolutional Neural Network Based Data-Driven Fault Diagnosis Method", *IEEE Transactions on Industrial Electronics*, vol. 65, no. 7, pp. 5990-5998, 2017.

[13] Abdenour Soualhi, Kamal Medjaher, and Noureddine Zerhouni, "Bearing Health Monitoring Based on Hilbert–Huang Transform, Support Vector Machine, and Regression", *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 1, pp. 52-62, 2014.

[14] Kezhi Mao, Peng Wang, Robert X Gao, Rui Zhao, Ruqiang Yan, and Zhenghua Chen, "Deep learning and its applications to machine health monitoring", *Mechanical Systems and Signal Processing*, vol. 115, pp. 213-237, 2019.

[15] Jinjiang Wang, Kezhi Mao, Rui Zhao, and Ruqiang Yan, "Learning to monitor machine health with convolutional bi-directional LSTM networks", *Sensors*, vol. 17, no. 2, pp. 273, 2017.

[16] Konstantinos Gryllias, Weihua Li, and Zhuyun Chen, "Mechanical fault diagnosis using convolutional neural networks and extreme learning machine", *Mechanical systems and signal processing*, vol. 133, pp. 106272, 2019.

[17] Hao Zhang, Qiang Zhang ,Siyu Shao, Tianlin Niu, and Xinyu Yang, "Attention-based LSTM network for rotatory machine remaining useful life prediction", *IEEE Access*, vol. 8, pp. 132188–132199, 2020.

[18] Kamran Javed, Noureddine Zerhouni, Patrick Nectoux, and Rafael Gouriveau, "Enabling health monitoring approach based on vibration data for accurate prognostics", *IEEE Transactions on industrial electronics*, vol. 62, no. 1, pp. 647-656, 2014.

[19] Kamran Javed, Noureddine Zerhouni, and Rafael Gouriveau, "State of the art and taxonomy of prognostics approaches, trends of prognostics applications and open issues towards maturity at different technology readiness levels", *Mechanical Systems and Signal Processing*, vol. 94, pp. 214-236, 2017.

[20] Brigitte Chebel-Morello, Christophe Varnier, Emmanuel Ramasso, Kamal Medjaher, Noureddine Zerhouni, Patrick Nectoux, and Rafael Gouriveau, "PRONOSTIA: An experimental platform for bearings accelerated degradation tests", *IEEE International Conference on Prognostics and Health Management, PHM*, pp. 1-8, 2012.

[21] Jianjing Zhang, Peng Wang, Robert X. Gao, and Ruqiang Yan, "Long short-term memory for machine remaining life prediction", *Journal of manufacturing systems*, vol. 48, pp. 78 - 86, 2018.

[22] Pierre Baldi, Ruqiang Yan, Stephen McAleer, and Siyu Shao, "Highly accurate machine fault diagnosis using deep transfer learning", *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2446–2455, 2018.

[23] Hesam Addin Arghand, Aref Aasi, and Abbas Rohani Bastami, "Estimation of remaining useful life of rolling element bearings using wavelet packet decomposition and artificial neural network", *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, vol. 43, no. 1, pp. 233-245, 2019.

[24] Kwok Leung Tsui, and Nan Chen, "Condition monitoring and remaining useful life prediction using degradation signals: Revisited", *IiE Transactions*, vol. 45, no. 9, pp. 939-952, 2013.

[25] Rui Zhao, Ruqiang Yan, Siyu Shao, Wenjun Sun, and Xuefeng Chen, "Convolutional discriminative feature learning for induction motor fault diagnosis", *IEEE Transactions on Industrial Informatics*, vol. 13, no. 3, pp. 1350-1359, 2017.

[26] Jun Zhu, Nan Chen, and Weiwen Peng, "Estimation of bearing remaining useful life based on multiscale convolutional neural network", *IEEE Transactions on Industrial Electronics*, vol. 66, no. 4, pp. 3208-3216, 2018.

[27] Ahmed Mosallam, Kamal Medjaher, and Noureddine Zerhouni, "Data-driven prognostic method based on Bayesian approaches for direct remaining useful life prediction", *Journal of Intelligent Manufacturing*, vol. 27, no. 5, pp. 1037-1048, 2016.

[28] Biao Wang, Naipeng Li, Ningbo Li, and Yaguo Lei, "A hybrid prognostics approach for estimating remaining useful life of rolling element bearings", *IEEE Transactions on Reliability*, vol. 69, no. 1, pp. 401-412, 2018.

[29] Jingqiang Zou, Xin Wang, and Yi Qin, "The optimized deep belief networks with improved logistic sigmoid units and their application in fault diagnosis for planetary gearboxes of wind turbines", *IEEE Transactions on Industrial Electronics*, vol. 66, no. 5, pp. 3814-3824, 2018.

[30] Peng Wang, Robert X Gao, Ruqiang Yan, Siyu Shao, and Yadong Lu, "DCNN-based multi-signal induction motor fault diagnosis", *IEEE Transactions on Instrumentation and Measurement*, vol.69, no. 6, pp. 2658-2669, 2019.

[31] Ilya Sutskever, Oriol Vinyals, and Quoc V Le, "Sequence to sequence learning with neural networks", *Advances in neural information processing systems*, pp. 3104–3112, 2014.

[32] Chuang Sun, Meng Ma, Ruqiang Yan, Xingwu Zhang, and Xuefeng Chen, "A deep coupled network for health state assessment of cutting tools based on fusion of multisensory signals", *IEEE Transactions on Industrial Informatics*, vol. 15, no. 12, pp. 6415-6424, 2019.

[33] P.S. Heyns, and S.A. Aye, "An integrated Gaussian process regression for prediction of remaining useful life of slow speed bearings based on acoustic emission", *Mechanical Systems and Signal Processing*, vol. 84, pp. 485-498, 2017.

[34] Chuang Sun, Meng Ma, and Xuefeng Chen, "Discriminative deep belief networks with ant colony optimization for health status assessment of machine", *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 12, pp. 3115-3125, 2017.

[35] Chau Yuen, Jenny Chen Ni Low, Lahiru Jayasinghe, Shuzhi Sam Ge, and Tharaka Samarasinghe, "Temporal convolutional memory networks for remaining useful life estimation of industrial machinery", *2019 IEEE International Conference on Industrial Technology (ICIT)*, pp. 915-920, 2019.

[36] Feng Yang, Mohamed Salahuddin Habibullah, Pin Lim, Sivakumar Nadarajan, Tianyou Zhang, and Zhao Xu, "Health index-based prognostics for remaining useful life predictions in electrical machines", *IEEE Transactions on Industrial Electronics*,vol. 63, no. 4, pp. 2633-2644, 2016.

[37] Giduthuri Sateesh Babu, Peilin Zhao, and Xiao-Li. Li, "Deep convolutional neural network based regression approach for estimation of remaining useful life", *International conference on database systems for advanced applications*, pp. 214-228, 2016.

[38] A A Bakir, A Hassan, M F A Hamid, and M Zaman, "Prediction of remaining useful life for mech equipment using regression", *Journal of Physics: Conference Series*, vol. 1150, no. 1, pp. 012012, 2019.

[39] Bing Li, Chuang Sun, Huihui Miao, and Jie Liu, "Joint learning of degradation assessment and RUL prediction for aeroengines via dual-task deep LSTM networks", *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 5023-5032, 2019.

[40] Alexander G Hauptmann, Boyuan Yang, and Ruonan Liu, "Simultaneous bearing fault recognition and remaining useful life prediction using joint-loss convolutional neural network", *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 87-96, 2019.

[41] G. Murali, Revathi Venkataraman, and S. Sharanya, "Analysis of Machine Learning Based Fault Diagnosis Approaches in Mechanical and Electrical Components", *International Journal of Advanced Research in Engineering and Technology (IJARET)*, vol. 11, no. 10, pp. 80-94, 2020.

APPENDIX: SOME IMPLEMENTATION RESULTS ARE EMBEDDED HERE

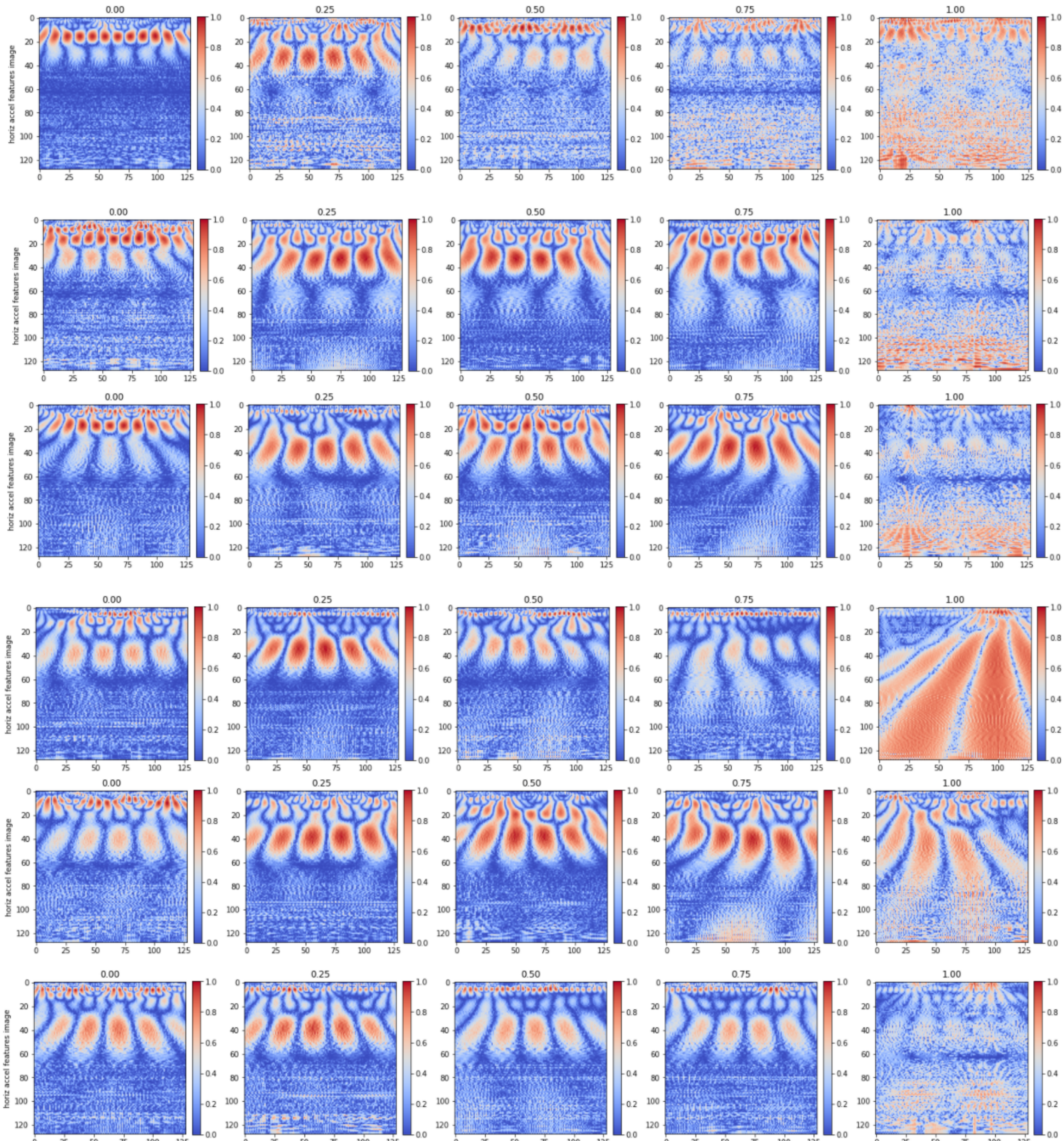


Fig. 10: Change in deterioration process of all 6 bearings horizontal acceleration vibration features in learning dataset

Fig. 11: Change in deterioration process of all 6 bearings vertical acceleration vibration features in learning dataset

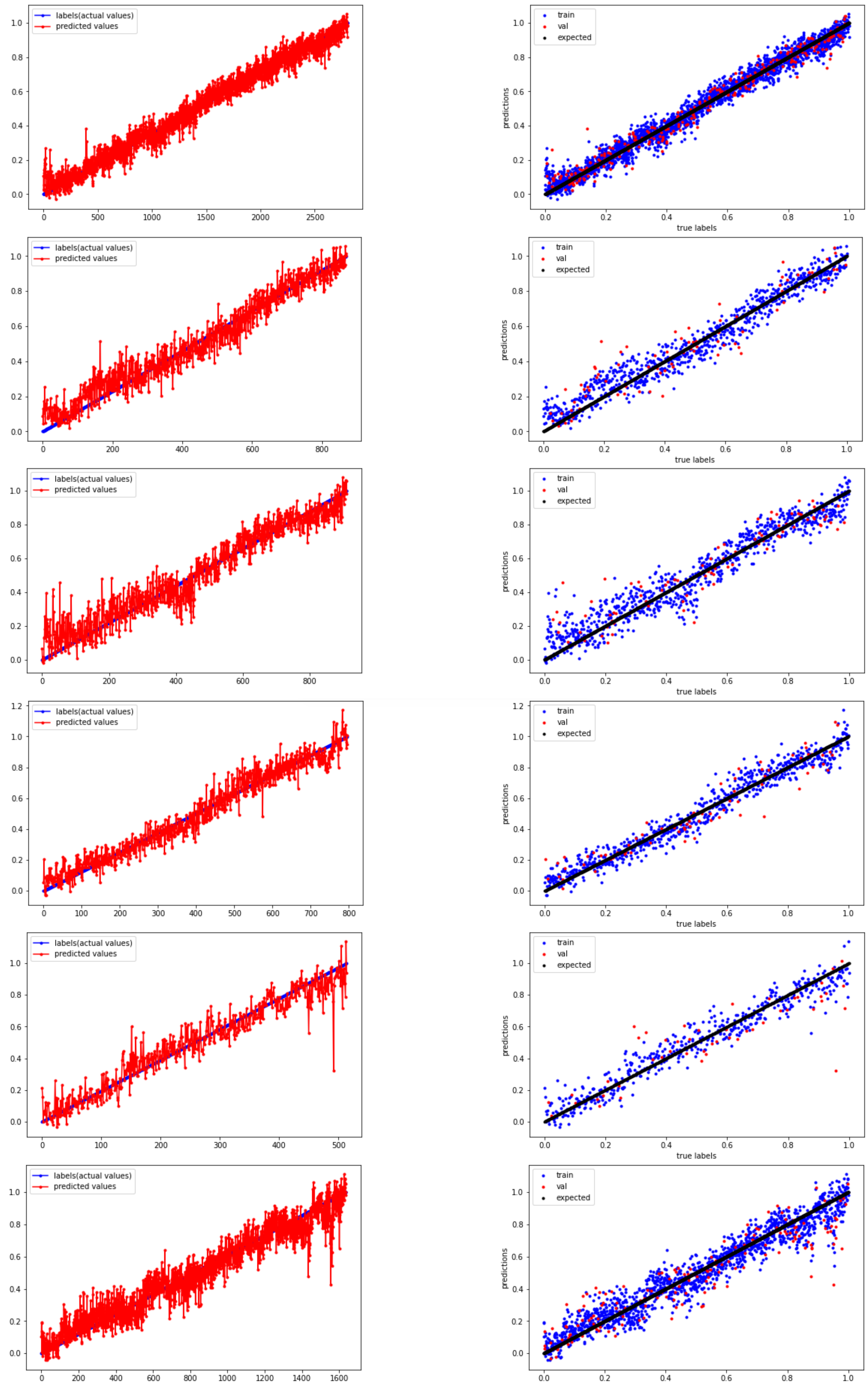


Fig. 12: CNN Model performance on train data and validation data of learning dataset

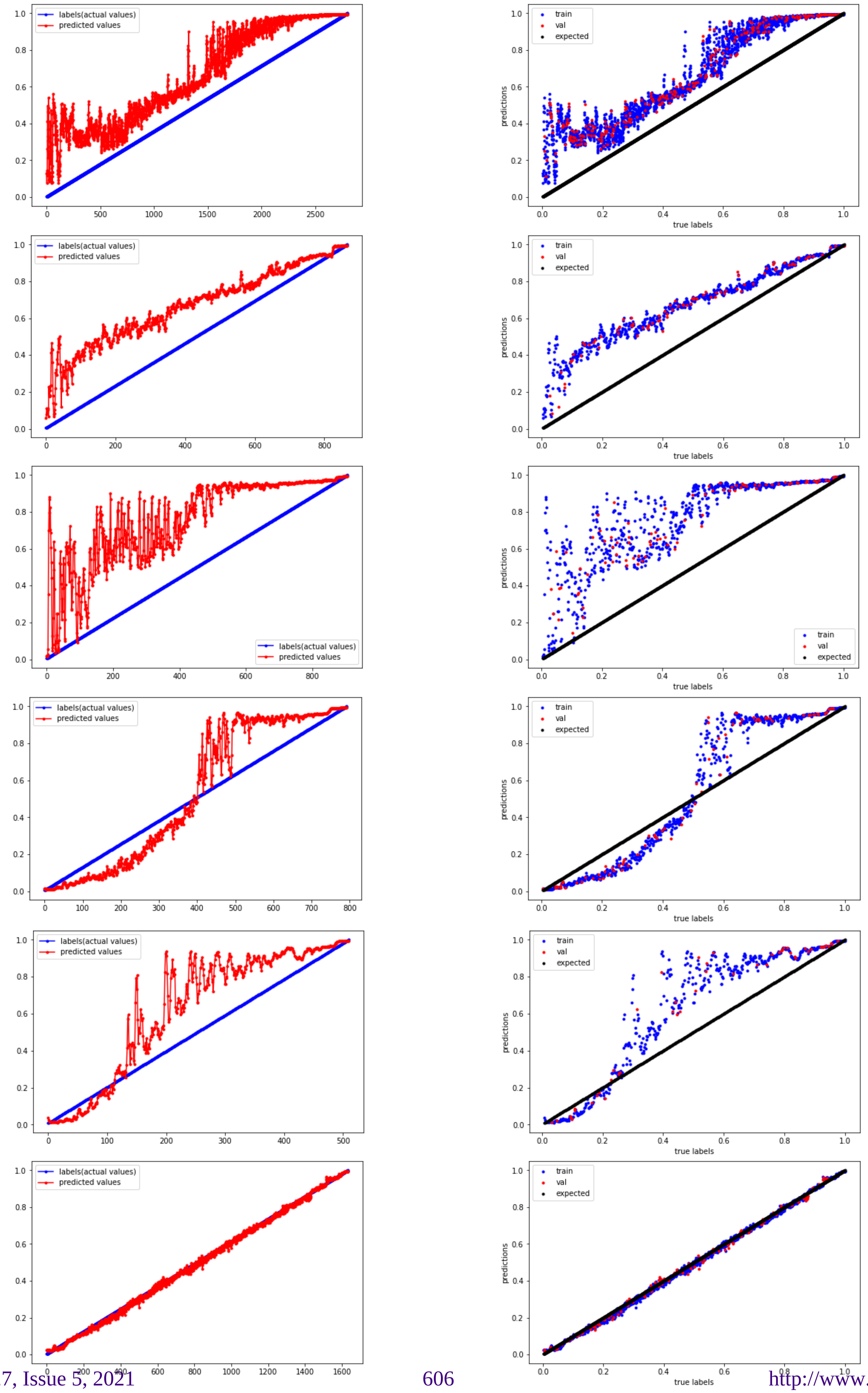


Fig. 13: CNN + LSTM performance on train data and validation data of learning dataset

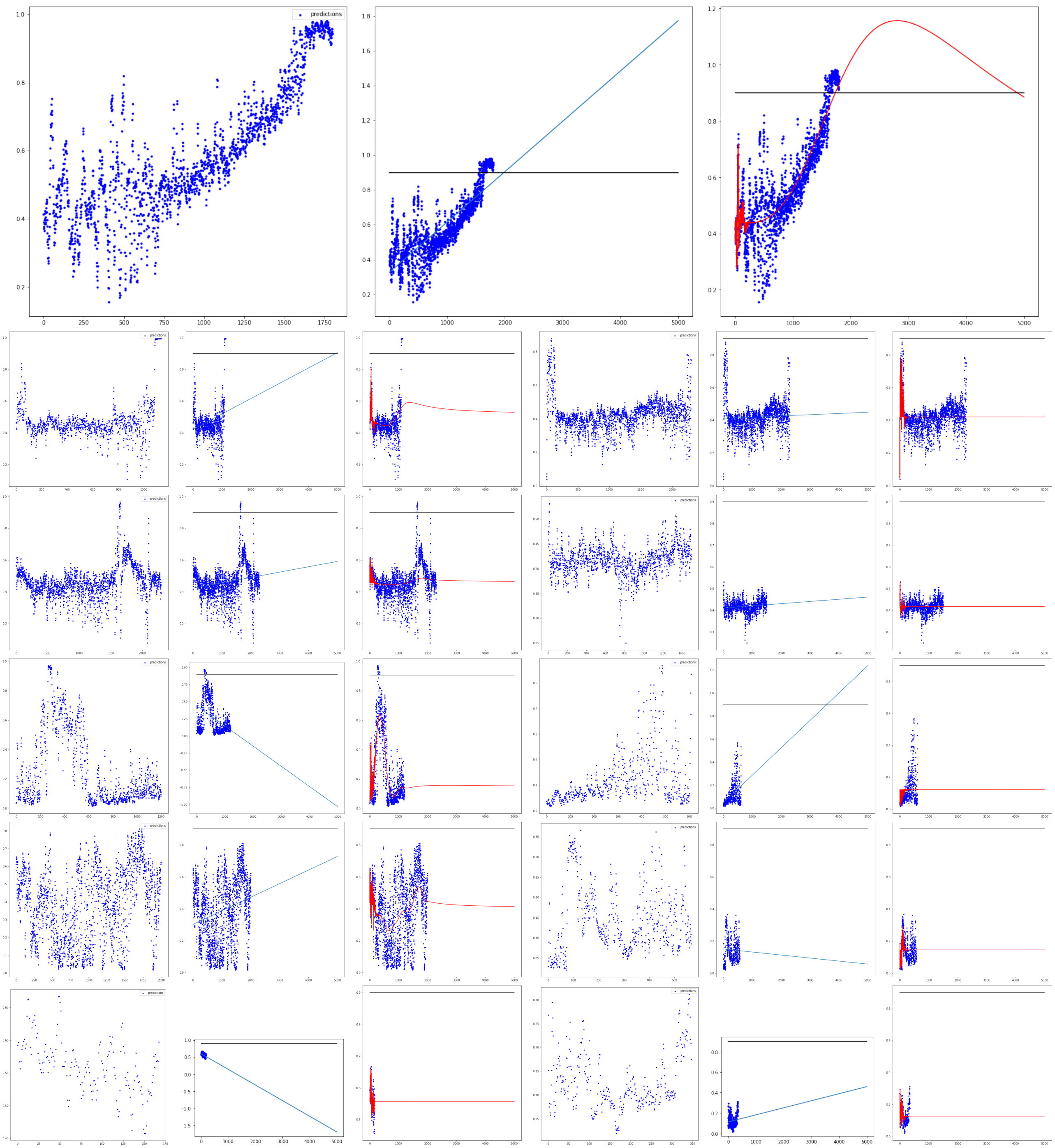


Fig. 14: CNN+LSTM results on test dataset