

Project Report

Explain the business use case.

Many Business use cases could be suggested due to recent advancements in Machine Learning and AI, but particularly it can be used for Predictive Maintenance in Automobiles, Production, Textiles, etc such Industries. The general concept remains for providing status update for machines for maintenance or even for accidental breakdowns, which will help organizations to early preparation for maintenance eventually avoiding losses.

Explain the technical implementation in short

The condition monitoring of Hydraulic Pump for Prediction provided by University of California; Irvine was used as training data. Since the dataset has Timestamp, Remaining Useful Life (RUL), was calculated by backtracking the time of actual breakdown, which now gives us samples and its time for breakdown. A regression model was developed for RUL prediction using the sensor data. A Classification model was developed for Cooler, accumulator, leakage, valve and stability failure prediction individually. Finally, all the models were loaded and passed through a template which takes in a single sensors data and print output from all the models in a templated format in form of early predictions to the end user.

Explain the package installation process, if you've used some commands to install various packages Prominently, Keras, TensorFlow and sklearn frameworks were used for Machine Learning and training, NumPy, and pandas were used for processing the data. Python was used as backend programming language. All the files and data set was saved on google drive and Google Colab with 11GB GPU was used as development environment.

Explain your input dataset format

The input data is curated from individual sensors data into a excel file with around 2205 samples. The sensors data was sampled at different rates like 1Hz, 10Hz and 100Hz. The final data is interpolated by up sampling and down sampling the individual sensors data with profile.txt which has actual targets. Eventually a dataset was curated and saved in excel format. However, the already curated dataset can also be obtained from the Kaggle website, which can be found [here](#). The complete intuition to this processing can also be found [here](#)

Explain your manual/automatic data pre-processing steps in detail

The RUL was calculated and processed from the Timestamp and backtracking the failure timestamp. Further Data pre-processing like dropping the Nan, null and infinite values was done. The dataset is splitted into training and testing with 80:20% split using sklearn library.

Explain your training pipeline in detail

Neural Network was used as ML architecture model for prediction. A Neural Network with 3 layers having 12 neurons in the input layer, 6 neurons in the hidden second layer and finally the corresponding classes as output layer. Adam and stochastic gradient descent (SGD) is used as optimizer with categorical cross entropy loss. The hyperparameters were determined after rigorous trial and error for obtaining optimal accuracy metric.

Explain how will you save the models for inference?

The trained model is saved in .h5 format using keras library in weights folder.

Explain the evaluation metric used to evaluate models

For classification purpose particularly sklearn library was used. Confusion matrix, accuracy, F1 score, precision and recall such evaluation metrics were used to evaluate the models.

Explain your code directory structure for all of the git branches

The weights folder contains the trained weights. The training folder contains program files for training the model and obtaining new weights. The Prediction. ipynb file can be used to directly get prediction from already trained weights.

If you've used open-source data, mention data link along with other required links

The open-source dataset link can be found [here](#).