March 2020

A FRAMEWORK TO MONITOR A FAILURE PREDICTIVE MACHINE LEARNING MODELS PERFORMANCE WITH NON-FAILURE DATA

HP INC

A FRAMEWORK TO MONITOR A FAILURE PREDICTIVE MACHINE LEARNING MODELS PERFORMANCE WITH NON-FAILURE DATA

Abstract

We built a new framework to evaluate a production failure predictive model performance using a set of healthy non failure data. This framework gives us the capability to quickly access the model performance as soon as we deploy it into production. In the past, in order to access a failure predictive model performance, ones need to have some number of failures to happened in the field and then looking back in time if prediction was ever made for those failures. If the true positive windows is 60 days then you have to wait more than 60 days to be able to access a field model performance. We can shorten that waiting period by leveraging healthy data that we collected from the field.

Keywords : Failure predictive model ; Model evaluation ; Performance

Problem Statements

A failure predictive machine learning model must give accurate predictions to create real value. We must have confidence the model is working as expected and trust the resulting predictions. Could the model perform well enough with unseen data? These issues can be handled by evaluating the performance of that model, such as precision and recall values. In order to do that, having field failure data is crucial. What if we don't have enough failure data? Can we leverage non-failure data to understand the performance of the model? Apparently, we can.

Our Solutions

A failure predictive model will predict a printer/device part failure within a predefined time window, such as 30 days, 60 days, etc. If failure data is available, we can use precision and recall evaluating the performance of a failure predictive model, as explained below. Precision (P) measures how well the model predicts a failure and Recall (R) measures how many failures the model gets correct. Mathematically, Precision is defined as a ratio of number True Positives (TP) to the sum of True Positives (TP) plus False Positives (FP) [1]:

$$P = \frac{TP}{TP + FP} , \qquad where \qquad (1)$$

$$TP + FP = all\ predictions \qquad (2)$$

Mathematically, Recall is defined as a ratio of True Positives (TP) to the sum of True Positives (TP) and False Negatives (FN) [1]:

$$R = \frac{TP}{TP + FN} , \qquad where \qquad (3)$$

$$TP + FN = all\ repairs \tag{2}$$

For example, a hypothetical model predefines TP windows as 30 days and makes a prediction for device “X001”. If a failure occurs on device “X001” within 30 days from then prediction, then the prediction is counted as TP. If the failure occurs after 30 days, then it is counted as FP. And finally, if the failure occurs before the prediction, then it is counted as FN. This approach strongly relies on the presence of failure data. If there is no failure data, these metrics cannot be calculated.

Our approach allows us to calculate the confidence level of a precision measure using non-failure data. Logically speaking, when the predictive failure model is exercised on a set of healthy (non-failure) data, the model should return no failure predictions, meaning we should not predict failures in healthy data. If the model returns a failure prediction, then it is counted as FP, because the model predicts a failure, but we know the data is healthy. For example, if we exercise a predictive failure model using data from 10 healthy devices and the model predicts failures in two devices, we have two counts of FP.

Another crucial thing that we need to do when we built the step in prediction training model training is to run a bootstrapping cross validation technique [2], [3]. Meaning that in addition to calculating the Precision (P) and Recall (R) during training, we also calculate the confidence interval (ΔP "or" ΔR).

$$P = P_0 \pm \Delta P \;\; => \;\; Precision \tag{5}$$

$$R = R_0 \pm \Delta R \;\; => \;\; Recall \tag{6}$$

Where Po and Ro are the mean precision and recall.

In our approach, we are assuming the model has high precision, which mean that the value of False Positive (FP) is small, hence the ratio of FP TP is much smaller compare to unity. Using Taylor expansion [4], [5], we can loosely show that ΔP ~ FP;

$$P = \frac{TP}{TP + FP} = \frac{1}{1 + \frac{FP}{TP}} \tag{7}$$

Assuming $\frac{FP}{FP} = x$; and $FP \ll TP$, then we can use Taylor expansion here:

$$P = \frac{1}{1 + x} \approx 1 - x = 1 - \frac{FP}{TP} \tag{8}$$

Multiplying both side with TP, we will end up with

$$TP * P = TP - FP \tag{9}$$

Hence if we compare Eq. 5 and Eq. 9, we can loosely assume that $FP \approx \Delta P$. Using these relationships, we can safely claim that if the number of FP in the healthy data predictions results is equal or smaller than ΔP from the training result, then we can claim that the model performance is about the same or better than training model performance.

Advantages

This approach will help us realize quickly how the model in production perform. We don't have to wait for failure data to be able to know the performance of the model. We can analyze the performance of a model on production before failure data is available.

References

[1] David M W Powers. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation In Journal of Machine Learning Technologies. 2 (1):, pages 37–63.

[2] B Efron and R Tibshirani (1993). An Introduction to the Bootstrap. In Boca Raton, FL: Chapman & Hall/CRC.ISBN 0-412-04231-2.

[3] Bradley Efron (2003). Second Thoughts on the Bootstrap.

[4] Milton Abramowitz and Irene A Stegun (1970). Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. In New York: Dover Publications, Ninth printing.

[5] George B Thomas and Ross L Finney Jr (1996). Calculus and Analytic Geometry. In Calculus and Analytic Geometry (9th ed.), Addison Wesley, ISBN 0-201-53174-7.

Disclosed by Manjunath Bhuyar, Sarah Bradburn, Prasad Hegde, Dan Siddall, Katy Ferguson and Anton Wiranata, HP Inc.