Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

EEP55C23
Computation for Transport Engineering

Anomaly Detection for Predictive Maintenance Applications using Auto-Encoders

Abhishek Dutta Gupta
ID : 22312353

Project report for Computation for Transport Engineering - 5 credit module.

2022-2023

Abstract

This project utilized a synthetic predictive maintenance dataset to develop an anomaly detection system for identifying failed machines. The dataset contained information on various parameters such as Product ID, air temperature, process temperature, rotational speed, torque, tool wear, machine failures, and different types of machine failures. The study focused on the importance of rotational speed and torque in detecting anomalies and used a simple autoencoder neural network to train the model. The results showed that the failed dataset had a higher mean squared error (MSE) than the reconstructed or test sets, indicating the success of the anomaly detection system. Overall, this study highlights the potential of synthetic predictive maintenance datasets and autoencoder neural networks for anomaly detection in manufacturing systems.

Project Objective

The project aims to employ an auto-encoder neural network to detect anomalies in a machinery dataset. This dataset comprises different features for various tools, with a label indicating whether the machine has failed and the type of failure. The autoencoder network is utilized to identify failed machines by calculating the reconstruction error metric for each sample. The primary aim is to demonstrate that this encoder's reconstruction error values will be higher for anomalous points, indicating faulty machines. Therefore, the autoencoder network will differentiate between good and bad classified data by identifying anomalies in the dataset.

What are auto-Encoders?

Auto-encoders [1] is a type of neural network that employs a trivial labeling approach by setting the output labels y to be the same as the input x. Therefore, auto-encoders aim to reconstruct the input as precisely as possible, making it an identity function algorithm. In easier terms 1, the input data (x1, x2, x3, x4) is attached to a hidden layer (z1, z2) and then rebuilt as ($\hat{x}1$, $\hat{x}2$, $\hat{x}3$, $\hat{x}4$).

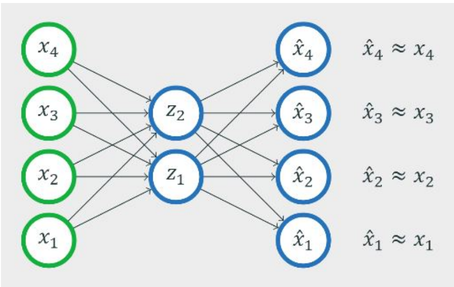

Figure 1: *Simple Auto-encoder Representation*

So, we keep on training the network to reduce the difference between the input and the reconstructed output, which is measured using mean squared error for our case. By doing so, auto-encoders can capture the underlying structure of the input data and identify patterns or anomalies in the dataset.

Technical Literature of the Dataset

In this project, we utilized a synthetic yet realistic predictive maintenance dataset provided by Stephen Matzka [2]. The dataset comprises several columns, including the Product ID, air temperature, process temperature, rotational speed, torque, tool wear, machine failures, and five types of machine failures: TWF, HDF, PWF, OSF, and RNF. For our project, we utilized Product ID as a unique identifier for each tool, while air temperature, process temperature, rotational speed, and torque were used as features. Additionally, the machine failures column was used as a label for classification, specifically for identifying failed machines. Below we briefly, describe the identifiers and features. A glimpse of the dataset is shown below.

	UDI	Product ID	Type	Air temperature [K]	Process temperature [K]	Rotational speed [rpm]	Torque [Nm]	Tool wear [min]	Machine failure	TWF	HDF	PWF	OSF	RNF
0	1	M14860	M	298.1	308.6	1551	42.8	0	0	0	0	0	0	0
1	2	L47181	L	298.2	308.7	1408	46.3	3	0	0	0	0	0	0
2	3	L47182	L	298.1	308.5	1498	49.4	5	0	0	0	0	0	0
3	4	L47183	L	298.2	308.6	1433	39.5	7	0	0	0	0	0	0
4	5	L47184	L	298.2	308.7	1408	40.0	9	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9995	9996	M24855	M	298.8	308.4	1604	29.5	14	0	0	0	0	0	0
9996	9997	H39410	H	298.9	308.4	1632	31.8	17	0	0	0	0	0	0
9997	9998	M24857	M	299.0	308.6	1645	33.4	22	0	0	0	0	0	0
9998	9999	H39412	H	299.0	308.7	1408	48.5	25	0	0	0	0	0	0
9999	10000	M24859	M	299.0	308.7	1500	40.2	30	0	0	0	0	0	0

Figure 2: *Predictive maintenance Dataset*

- Product ID: This consists of unique serial number identification for each tool, additionally having the type of machine as L/M/H being low, medium, and heavy usage.
- Air Temperature and Process Temperature: This is the air temperature around the machinery, for this synthetic dataset, a value with a standard deviation of 2K around 300k is kept and designed. For process temperature, a standard deviation of 1K, with 10 added to air temperature is considered by Matzka.
- Rotational speed and Torque: The speed of the machinery is calculated in rpm and torque in Nm.
- Tool wear: This adds 2/3/5 minutes depending upon L/M/H kind of machinery.
- Machine Failure: This says, whether the tool has failed in this particular data point or not
- TWF, HDF, PWF, OSF, RNF: Different kinds of tool failures, if one of these failures are there, then machine failure is flagged.

Data Analysis and Trends

We have a total of 10000 data points X 14 columns. First, we analyze 3 the different kinds of machines we have, dividing and plotting them into Low, medium, and high. We notice a total 0f 6000 data points being of low type, with medium and high being the rest.

Next, we plot 4 the air temperature and the process temperature across all the data points, with failure points being marked, to see how the failed machine is impacted throughout. We notice that the failed machine points are distributed throughout, with anomalies not being related in any way to process or air temperature.

Similarly, we plot 5 for rotational speed and 6 for torque, with failed points being marked. Here, however, we notice some kind of pattern with failed points being more distributed towards high and low readings of rotational speed and torque. Thus we can say, rotational speed and torque and more important than air or process temperature in detecting anomalies, moreover, this is backed by Matzka's paper [2] where he concluded rotational speed and torque being important than air and process temperature through Estimated relative predictor importance of the features used in a graph.

Finally, we plot 7 a bar graph, with different kind of failures being analyzed HDF, PWF, and OSF being more.

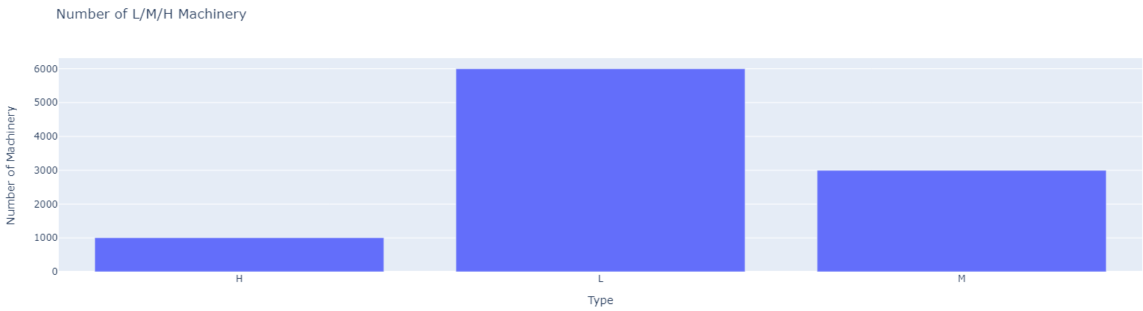

Figure 3: *Number of L/M/H Machinery*

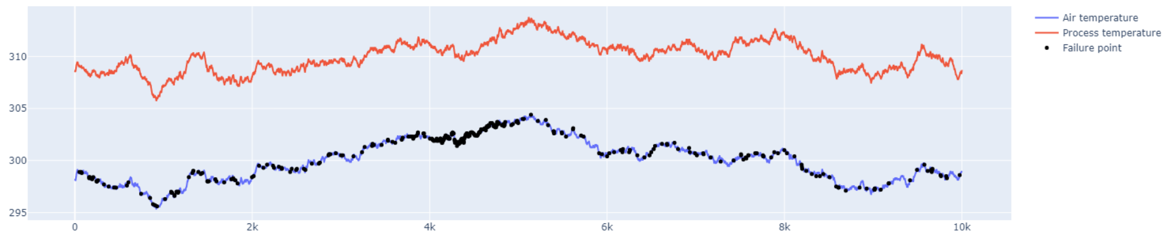

Figure 4: *Air Temperature and Process temperature throughout with failed points*

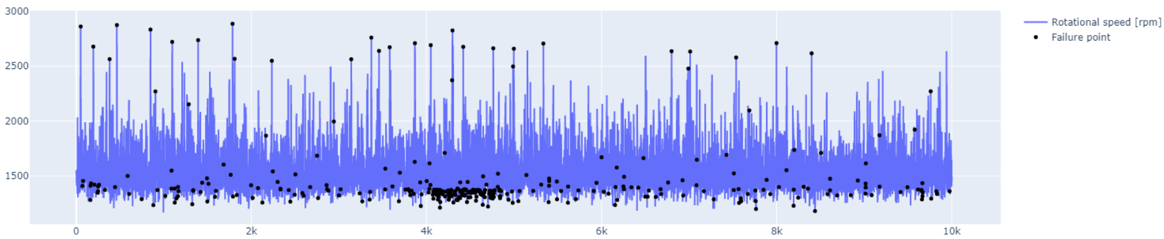

Figure 5: *Rotational speed throughout with failed points*

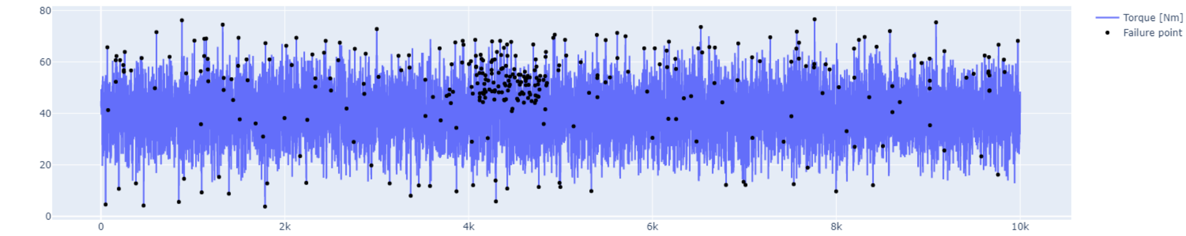

Figure 6: *Torque throughout with failed points*

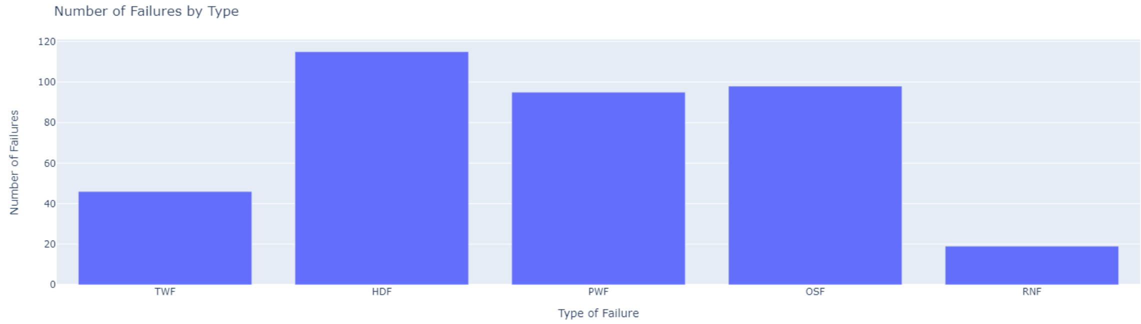

Figure 7: *Different kinds of failures*

Dataset preparation for the Encoder network

As explained before, we drop the other columns since only rotational speed and torque are important features for anomaly detection.

Next, the dataset is finally split into a safe dataset and a fail dataset, with the safe dataset containing the values with machine failure as 0 and the fail data set containing values with machine failure as 1.

The safe dataset is further split into a 75 percent training dataset and a 25 percent test or validation data set.

The Auto-encoder neural network

A simple autoencoder neural network is used for this project. A dense neural network 8 with three layers, consisting of one hidden layer. The input and output reconstructed layers consist of 100 units each and 30 units in the hidden layer, with an additional reshaping layer in the output. The Relu activation function and Adam optimizer are used. The network summary and the representation below explain the same.

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_9 (Dense)             (None, 100)               400

 dense_10 (Dense)            (None, 30)                3030

 dense_11 (Dense)            (None, 100)               3100

 dense_12 (Dense)            (None, 3)                 303

=================================================================
Total params: 6,833
Trainable params: 6,833
Non-trainable params: 0
_________________________________________________________________
```

Figure 8: *Summary of the dense autoencoder*

The model is then trained with 75 percent of the safe dataset created and then tested against the 25 percent of the safe dataset, the reconstructed new safe dataset, and the failed dataset. The model 9 loss is shown below. As per the objective, the failed dataset should have a higher MSE compared to the safe test set and reconstructed safe set.

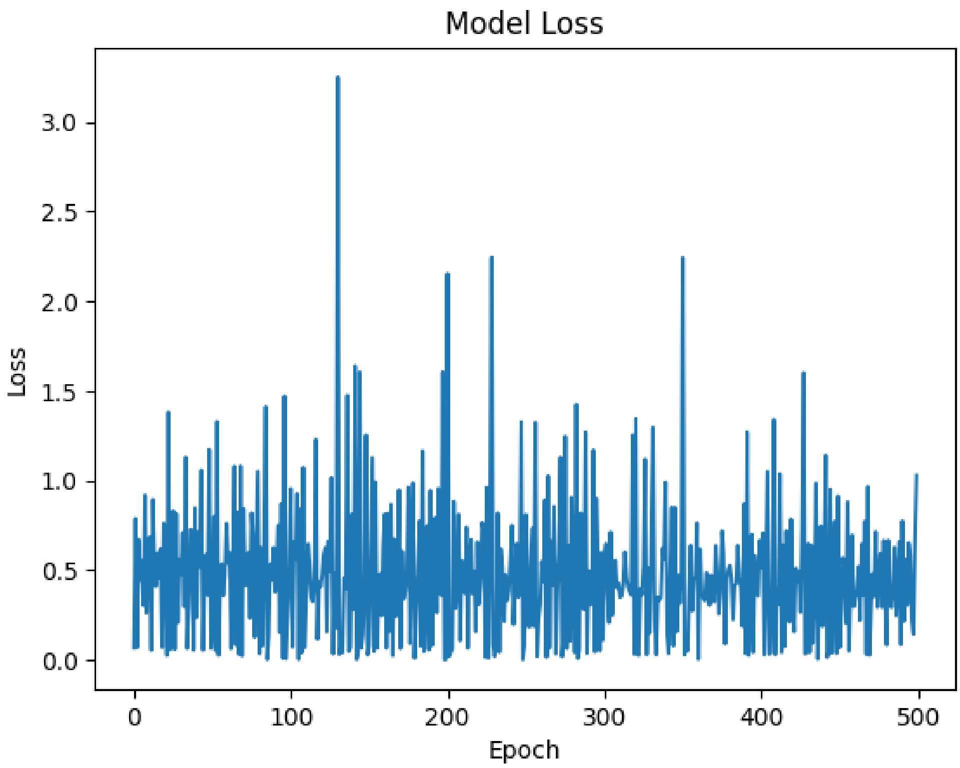

Figure 9: *Trained Model loss*

Results

Mean squared error is used as the metric for anomaly detection. The MSE is calculated for the reconstructed dataset, test set, and failed data set. The results for 500 epochs show that, collectively the failed dataset 10 has a higher MSE than the reconstructed or the test set, which confirms the case for anomaly detection.

Finally, individually all the data points are predicted with these metrics, where we also see, the failed machinery points having higher MSE to confirm the case. The black lines 11 represent the failed points.

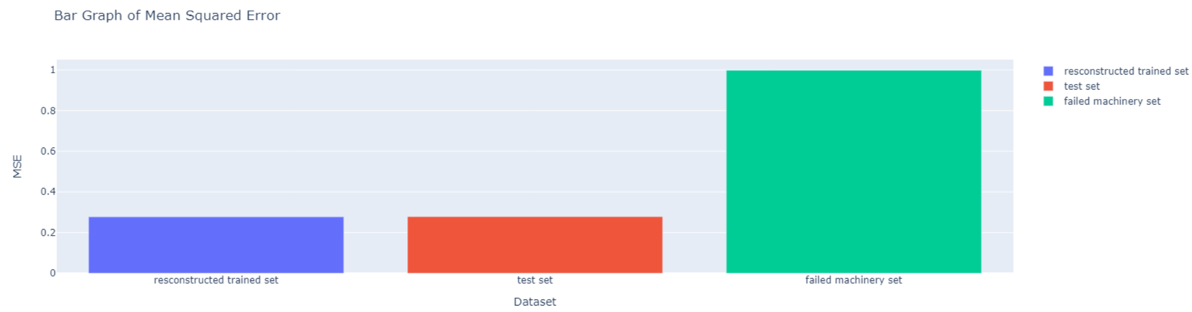

Figure 10: *Collective MSE for reconstructed, test and failed datasets*

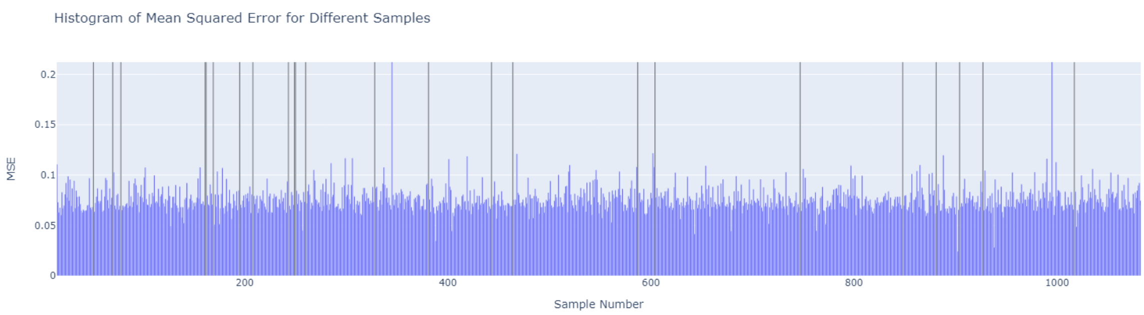

Figure 11: *Individual Sample MSE (blue lines = Normal points, black lines = failed points)*

Extra Analysis - including Tool Wear as classification

Finally, in addition, Tool wear was only tried as a feature for detecting anomalies. The same network is used, with TWF being used instead of the machine failure column. Here we notice similar results 12 with more MSE for the failed set. However, the MSE is less when Tool wear is included rather than when tool wear is not included, this further supports the fact of rotational speed and torque are more impactful in anomaly detection.

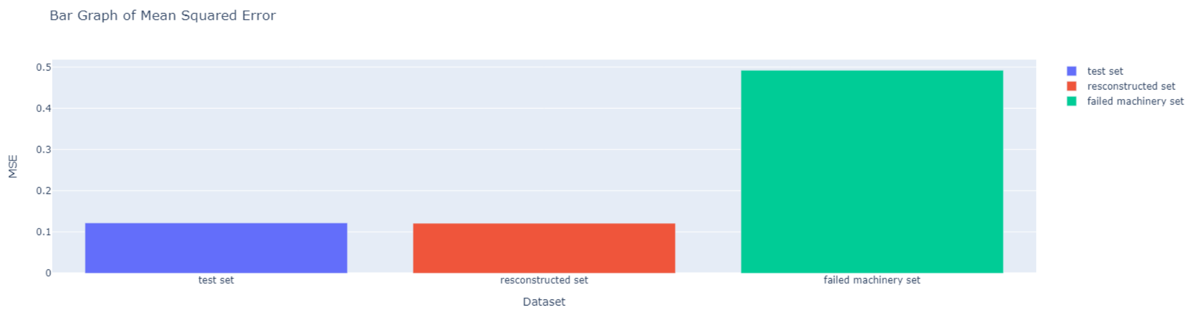

Figure 12: *MSE with Tool wear included in the network*

Future works

The Future works for this project could be to redo the network using an LSTM instead of a normal autoencoder network. Moreover, individual fail profiles could be made from the different kinds of failures available in the dataset, to see which kind of failure affects anomaly detection more.

1 Declaration

I have read and I understand the plagiarism provisions in the General Regulations of the University Calendar for the current year, found at http://www.tcd.ie/calendar. I certify that this submission is my own work.

References

[1] Kramer, M. A. Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal 37*, 2 (1991), 233–243.

[2] Matzka, S. Explainable artificial intelligence for predictive maintenance applications. In *2020 third international conference on artificial intelligence for industries (ai4i)* (2020), IEEE, pp. 69–74.