Motor maintenance prediction

Ji-Eun Lee

Research Assistant, Hershey Digital Innovation Lab

Outline

- Motivation
- Problem statement
- State-of-the-art approach
- Dataset
- Challenges
- Proposed approach
- Process
- Results
- Future work
- Reference

Motivation

Hershey has many factories that manufacture products

If a motor breaks down, this will cause immense loss

Can we create a model that sends out warnings requesting maintenance, when suspicious behavior was detected?

This can prevent loss caused by sudden motor failure

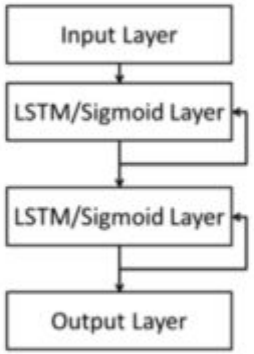
Problem statement

Build a model that can send different levels of warnings for maintenance request, especially predict imminent failure at least 12 hours prior to motor fault.

State-of-the-art approach

LSTM-AD

- Use stacked LSTM networks for anomaly/fault detection in time series.
- Three phases
    - Training
        - Train network with non-anomalous data to predict next consequent timesteps(for each feature)
        - Fit multivariate gaussian distribution for errors of the predictions
    - Validation
        - Using both non-anomalous and anomalous data, find the adequate threshold for confirming a time step as anomalous
    - Test
        - Use the model as an anomaly detector



State-of-the-art approach

<u>Training Phase</u>

- Train network with non-anomalous data to predict next consequent timesteps(for each feature)
- Fit multivariate gaussian distribution for errors of the predictions

e.g. Assuming there is only one feature, and prediction window = 2

At timestept, target value x_t

...

$x_{t-2} \rightarrow p_{t-1, 2}\ p_{t, 1}$

$x_{t-1} \rightarrow p_{t, 2}\ p_{t+1, 1}$

…

Notice that x_t has two predicted values (depends on the prediction window)

Error vector for $x_t = e_t = [|x_t - p_{t, 1}|, |x_t - p_{t, 2}|]$

State-of-the-art approach

Validation Phase

- Validation set consists of both non-anomalous and anomalous data
- Compute error vectors
- Find threshold that maximizes the F beta score
    - If the likelihood to observe a error vector is lower than some threshold classify as anomalous
- Since anomalous samples are still needed to learn the threshold for classifying an input as anomaly LSTM-AD is not zero-positive learning.

Test Phase

- Use the model as an anomaly detector

Dataset

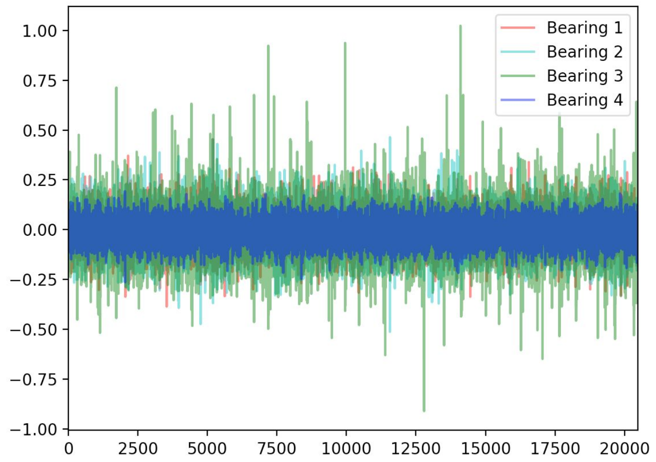
IMS Bearing Data

2000 RPM = 33.33 Hz
Rexnord ZA-2115 double row bearings (Four bearings)
PCB 353B33 High Sensitivity Quartz ICP accelerometers were installed on the bearing housing

680+ citations for original paper that gathered and used this dataset

Each recording for 1 second, with sampling rate set at 20 kHz (20,480 points)
every 10 minutes

Example recording (2004.02.12.10.32.39)



Dataset

3 sets of timeframes given,
each is a recording of motor acceleration data for four motors, until motor failure (Test-to-Failure)

1st dataset :
Four bearings acceleration recorded at both x and y axis = total 8 features.
However disregarded due to discontinuous recording

2nd and 3rd dataset :
Four bearings at single axis = total 4 features

Challenges

1. Anomaly detection problem
2. Dataset characteristic
    a. Preprocessing accelerometer data
    b. Test-to-failure (Only one anomaly)
3. How to decide threshold for different levels of warning

Proposed approach

[Train Phase]

Train the model using only normal data, so that the model learns the distribution of normal data points.

- We assume that in the beginning of the cycle, motor is going to display normal behavior

Using timestamps [t-N, ..., t], predict timestamps [t+1,...,t+M]

Compute anomaly score by computing the error vector between predictions and real data and fit the error vector to a multivariate Gaussian distribution to calculate log likelihood of observing this error.
This shows how likely it is to observe the error, considering the distance from the center of the fitted gaussian distribution.

- Compute mean (μ) and covariance (Σ) of error vectors
- Compute log likelihood using following equation
  x = error vector of consideration
  k = number of variables

$$\ln L = -\frac{1}{2}\left[\ln(|\boldsymbol{\Sigma}|) + (\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}) + k\ln(2\pi)\right]$$

Proposed approach

[Test Phase]

Using the trained model and fitted gaussian distribution, compute predictions and likelihood for test dataset. Since the trained model is based on normal data points, anomalous data points will have a higher error, hence leading to lower likelihood.

Based on the likelihood of observing the error vector, have thresholds for classifying "Normal (green)", "Suspect (yellow)", "Imminent Failure (red)" states.

Process

(Experiment 1) Initial prototype

- Average each one second recording for representation
- Retrain for each dataset
- Neural Network with fully connected layers
- Using datapoint at timestamp t, predict timestamp t+1

(Experiment 2) Upgrade model structure

- LSTM model structure
- Using timestamps [t-N, ..., t] predict timestamps [t+1,...,t+M]

(Experiment 3) Decide threshold

- Focusing on generalizability
- Drop larger than τ : **warning**
- After three consecutive drops larger than τ : **red alert** (imminent failure)

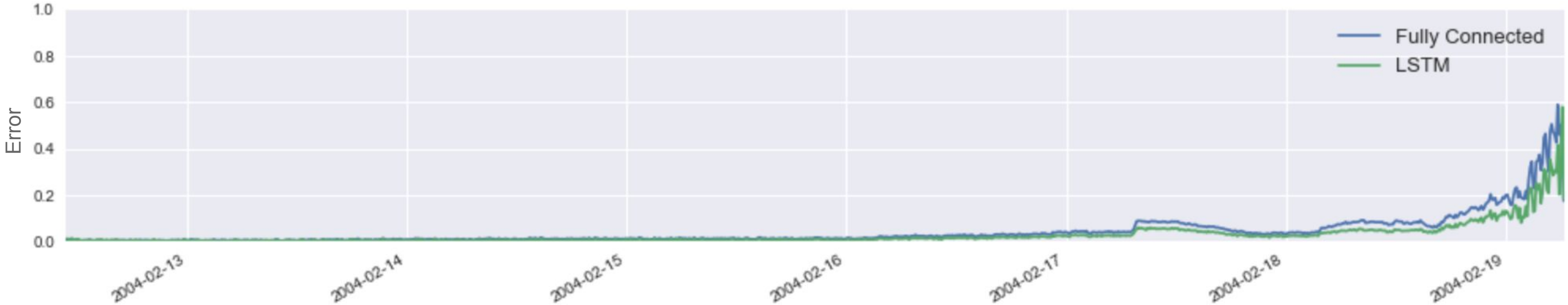
(Experiment 4) Use previously trained model to another dataset

[Experiments 1 and 2] Initial prototype → Upgrade model structure

Average each one second recording for representation
Retrain for each dataset
Neural Network with fully connected layers
Using datapoint at timestamp t, predict timestamp t+1

→

LSTM model structure
Using timestamps [t-N, ..., t] predict timestamps [t+1,...,t+M]



Error of prototype model = **1.6** x Error of LSTM model
LSTM is better suited for time series data

[Experiment 3] Decide threshold

Focusing on generalizability
Drop larger than τ : **warning**
After three consecutive drops larger than τ : **red alert** (imminent failure)

Dataset 2 (Feb 12 - Feb 19 : one week) : *threshold τ = -1000*

Warnings at **2 days, 19 hours** prior motor fault

Predicted Imminent Failure **13 hours 20 minutes** prior motor fault

[Experiment 3] Decide threshold

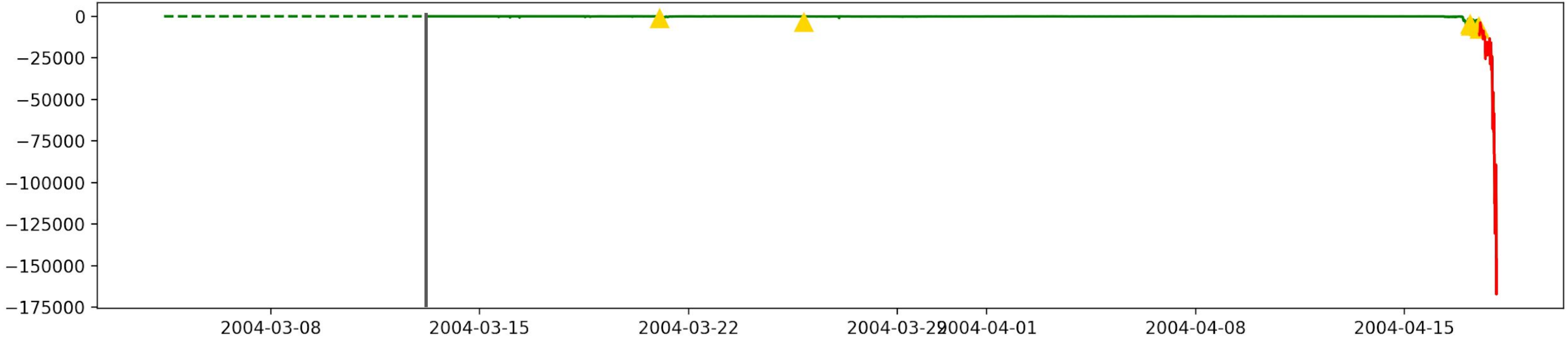
Dataset 2 (Feb 12 - Feb 19 : one week) : *<u>Using different threshold τ = -500</u>*

Predicted Imminent Failure 2 days prior motor fault, which is too early

- <u>finding the right threshold is a difficult task</u>

[Experiment 3] Decide threshold

Dataset 3 (Mar 4 - Apr 18 : six weeks) : *threshold τ = -1000*

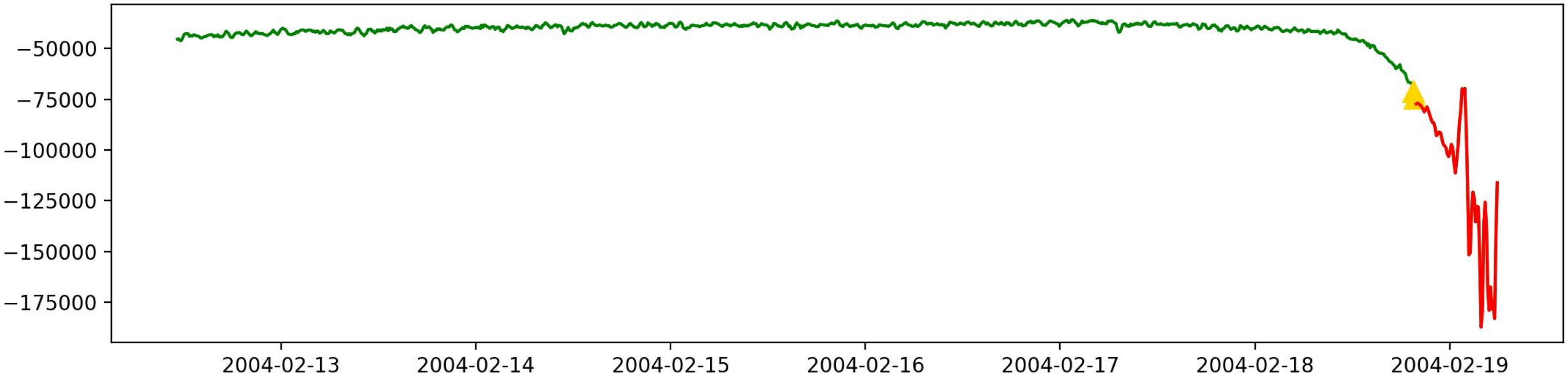


Warnings at **23 days, 22 hours** prior motor fault

Predicted Imminent Failure **14 hours 30 minutes** prior motor fault

[Experiment 4] Use previously trained model to another dataset

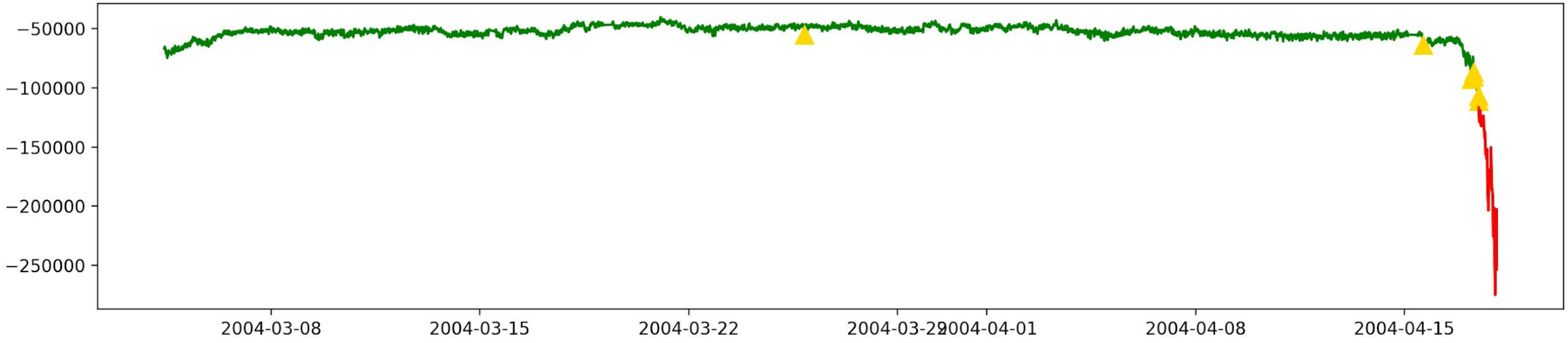
Dataset 2 (Feb 12 - Feb 19 : one week) *threshold τ = -3000*



Predicted Imminent Failure **10 hours 30 minutes** prior motor fault

[Experiment 4] Use previously trained model to another dataset

Dataset 3 (Mar 4 - Apr 18 : six weeks) *threshold τ = -4000*



Warnings at **2 days 12 hours, 20 hours**

Predicted Imminent Failure 15 hours prior motor fault

[Experiment 4] Use previously trained model to another dataset

Issues

- likelihood values start from -40000, -50000
- Can infer that the motor behavior and statistics change in a new set of motors

- However would be meaningful to be able to use previously trained model, without retraining after maintenance (future work!)

Future work

- Using autoencoder to extract representative vector for each one second recording

- Deriving threshold automatically

- Making the model real time
    - Real time normalization
        - Currently normalizing the entire dataset at the beginning, but that won't be possible in real-settings
    - Find adequate amount of training set
        - Currently using 20% of the dataset as training
        - Fixed number of days, online training, ...

References

J. Lee, H. Qiu, G. Yu, J. Lin, and Rexnord Technical Services. 2007. IMS, University of Cincinnati. "Bearing Data Set", NASA Ames Prognostics Data Repository (http://ti.arc.nasa.gov/project/prognostic-data-repository), NASA Ames Research Center, Moffett Field, CA

P. Malhotra, L. Vig, G. Shroff, and P. Agarwal. 2015. Long Short Term Memory Networks for Anomaly Detection in Time Series. In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*. 89–94.

T. Lee, J. Gottschlich, N. Tatbul, E. Metcalf and S. Zdonik. 2018. Greenhouse: A Zero-Positive Machine Learning System for Time-Series Anomaly Detection. https://arxiv.org/abs/1801.03168/. In *SysML Conference*.