Machine Learning & AI

September 18, 2025

TSX: PPL; NYSE: PBA

Umeet Bhachu

PEMBINA

About the Presenter

Umeet Bhachu, P.Eng, PMP, CMRP, Staff Engineer, Pembina Pipelines Corporation

Experience in rotating equipment and turbomachinery, with a Bachelor’s degree in Chemical Engineering from the University of British Columbia and a Mechanical Engineering Diploma from the British Columbia Institute of Technology. My career has been focused largely in oil and gas plants, specializing in rotor dynamics, turbo compressors, gas and steam turbines, pumps, and mechanical seals design and troubleshooting. Been with Pembina as a Staff Rotating Equipment Engineer for over 7 years, where I’ve also been actively involved in applying machine learning, artificial intelligence, and condition monitoring to both rotating machinery and process assets. Have presented at various technical conferences and authored multiple technical papers throughout my career.

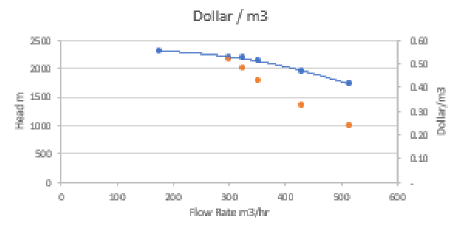
Condition Monitoring, Data Analytics, Advanced Analysis, Machine Learning and Deep Learning

- Learn to distinguish between basic condition monitoring – use of ASPEN, PI Vision etc. to monitor equipment. E.g. cases what pump is out of the curve operation, heat exchanger fouling, etc.
- Data analytics – Utilize averages, means, statistical data analysis, use of historical data to understand trends. E.g. cases what is a process doing is out control, SPC, making sense out of data.
- Advanced Data Analysis - focuses on diagnostics, predictive and prescriptive analysis of data. This includes optimization, linear/regression analysis, forecasting. This is a deeper dive into data analysis. E.g. Regression utilized to understand data deeper, causality vs correlation, helps in feature engineering.
- Machine and Deep Learning (AIs) – Predictions, learning, generating (LLMs like ChatGPT), Controlling and Discovering….. Ever developing field……. AGENTS.

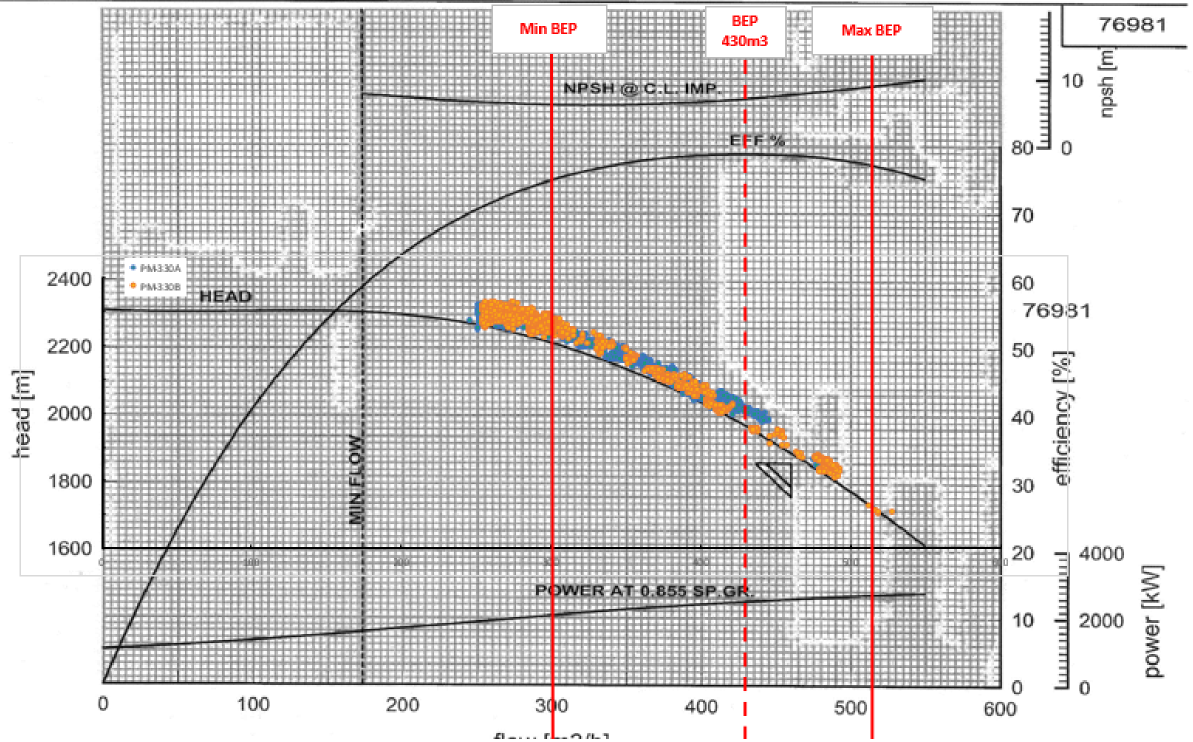
Condition monitoring - 2018

Date	#####	BEP Min Fl	301	m3/hr	Total Monthly Volume Pushed	m3	37560
	#####	BEP Flow	430	m3/hr	Cost/KWh	0 $/kWh	40632
		BEP Max F	516	m3/hr	Average SG	#	78192

		Flow [m3/h]	Head [m]	kWh/m3	Cost/m3	Monthly	Discrepancy	early Discrepancy
0.6	60%	175	2300	11.35	0.63	$ 49,542.37	$ 24,211.14	$ 290,533.68
0.7	70%	300	2200	9.31	0.52	$ 40,618.59	$ 15,287.36	$ 183,448.31
1	BEP (100%)	430	1960	5.81	0.32	$ 25,331.23	$ -	$ -
1.2	120%	516	1720	4.25	0.24	$ 18,524.54	$ (6,806.69)	$ (81,680.29)
0.75258	Avg. PM-330A	323.61	2182.21	8.59	0.48	$ 18,001.64	$ 5,336.02	$ 64,032.27
0.81741	Avg. PM-330B	351.49	2129.08	7.72	0.43	$ 17,492.70	$ 4,827.09	$ 57,925.03
	Combined				0.45	$35,494.34	$ 10,163.11	$ 121,957.30



Feb-19 - Mar-19	PM-330A		<70%	44.00%	5-12	56.00%		>120%	0.00%
	Hours	######		51.07		65.00			0.00
	PM-330B		<70%	50.98%	5-12	48.73%		>120%	0.29%
	Hours	######		58.93		56.33			0.33



Chat GPT is what many see as AI

[Submitted on 12 Jun 2017 (v1), last revised 2 Aug 2023 (this version, v7)]

Attention Is All You Need

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks in an encoder-decoder configuration. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

Comments: 15 pages, 5 figures
Subjects: **Computation and Language (cs.CL)**; Machine Learning (cs.LG)
Cite as: arXiv:1706.03762 **[cs.CL]**
(or arXiv:1706.03762v7 **[cs.CL]** for this version)
https://doi.org/10.48550/arXiv.1706.03762

Machine Learning

Machine learning (ML) is a type of Artificial Intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values. CAN GET ADVANCED VERY QUICKLY and relies on strong mathematically modelling and programming.

Types of machine learnings

Supervised, Un-supervised, Reinforcement

Supervised

Labelled data is provided with defined variables for the algorithm to assess for correlation.

- **Binary classification:** Dividing data into two categories.
- **Multi-class classification:** Choosing between more than two types of answers.
- **Regression modeling:** Predicting continuous values.
- **Ensembling:** Combining the predictions of multiple machine learning models to produce an accurate prediction.

Unsupervised

Training of the model is done on un-labelled data.

- **Clustering:** Splitting the dataset into groups based on similarity.
- **Anomaly detection:** Identifying unusual data points in a data set.
- **Association mining:** Identifying sets of items in a data set that frequently occur together.
- **Dimensionality reduction:** Reducing the number of variables in a data set.

Machine Learning

Semi Supervised

Mix of the first two types, part data is labelled, and other part is unlabeled.

- **Machine translation:** Teaching algorithms to translate language based on less than a full dictionary of words.
- **Fraud detection:** Identifying cases of fraud when you only have a few positive examples.
- **Labelling data:** Algorithms trained on small data sets can learn to apply data labels to larger sets automatically.

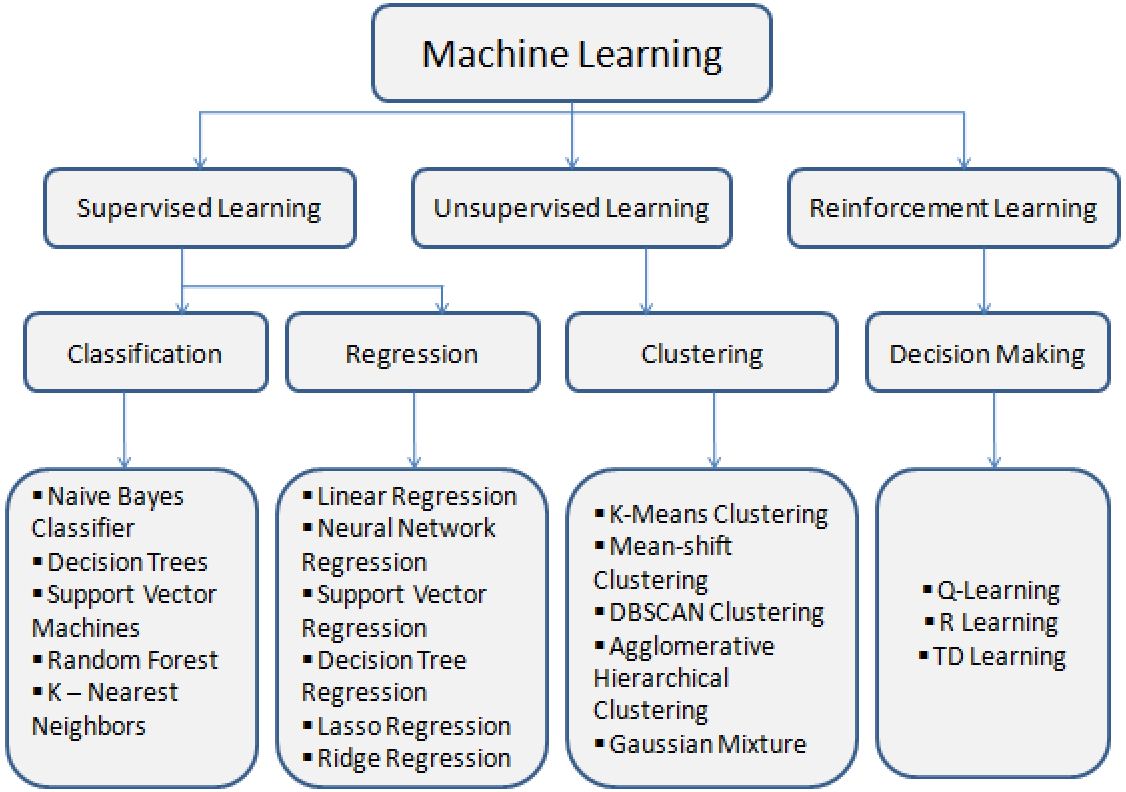
Reinforcement Learning

Teach a machine to complete a multi-step process for which there are clearly defined rules. An algorithm is programmed to complete a task based on positive or negative cues. But for the most part, the algorithm decides on its own what steps to take along the way.

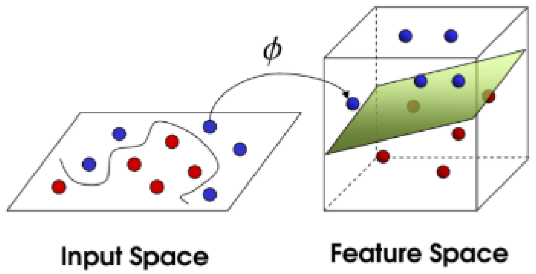
- **Robotics:** Robots can learn to perform tasks the physical world using this technique.
- **Video gameplay:** Reinforcement learning has been used to teach bots to play a number of video games.
- **Resource management:** Given finite resources and a defined goal, reinforcement learning can help enterprises plan out how to allocate resources.

Types of ML Algorithms

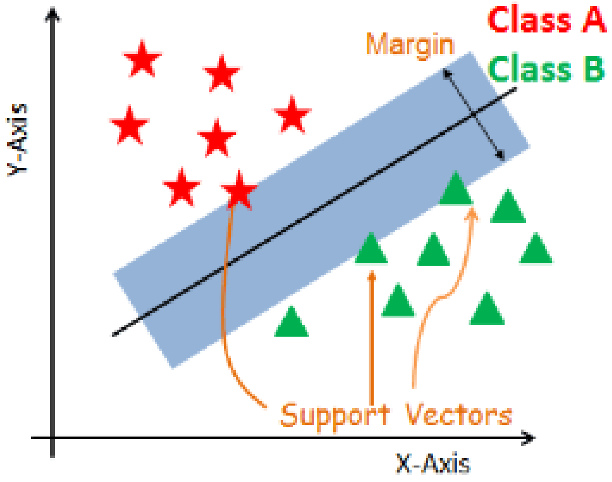
Litslink.com



Purpose of the Algorithm – Both Classification and Regression

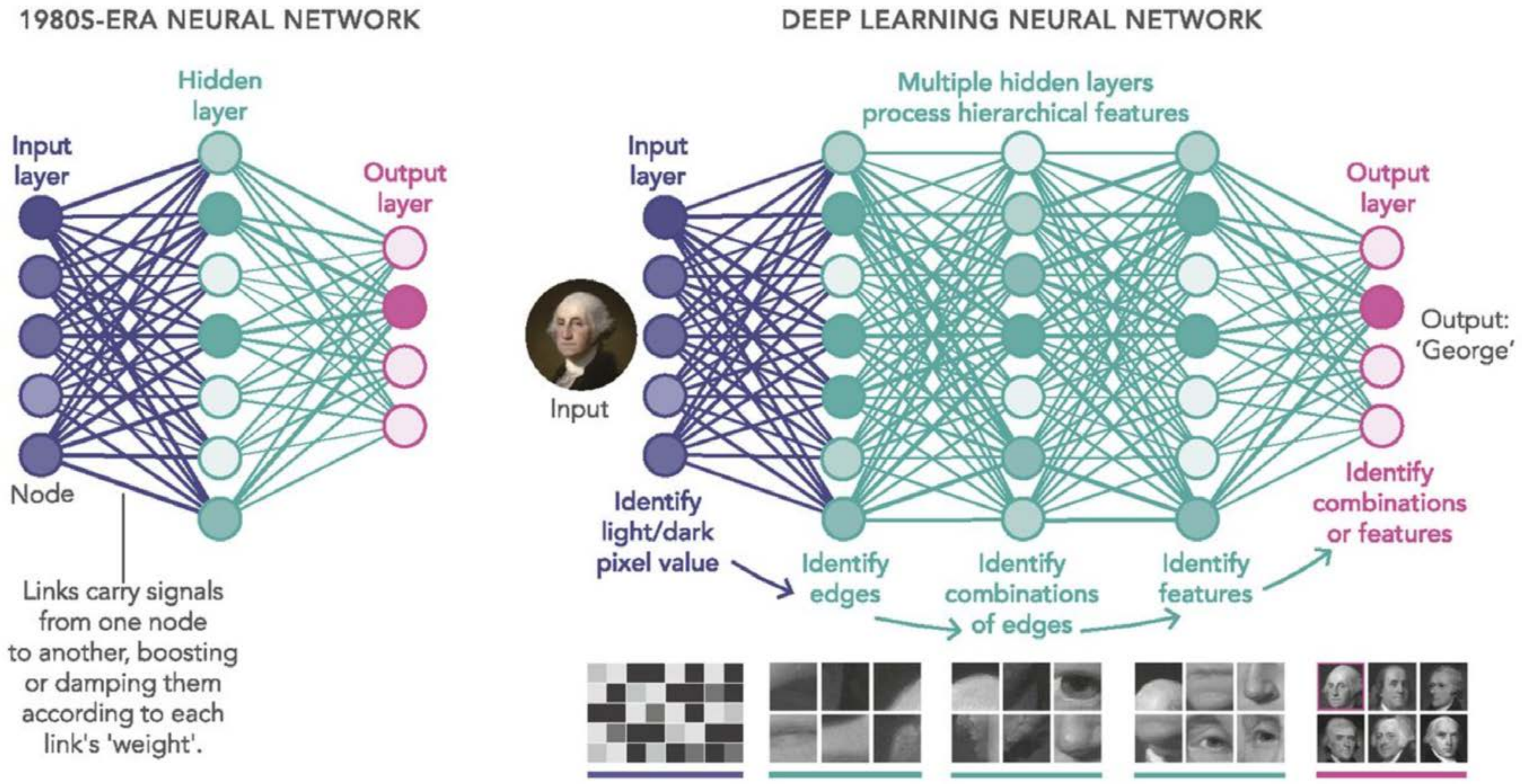


The goal of the algorithm involved behind SVM:



Last but not the Least – Deep Learning

Deep learning is a subset of machine learning , which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behaviour of the human brain—albeit far from matching its ability—allowing it to “learn” from large amounts of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy.



Support Vector Machine (SVM) VS Long Short-Term Memory (LSTM)

1. Whenever we are given any test feature vector x to predict, mapped to a complex Φ(x) and asked to predict which is basically $w^T\Phi(x)$
2. Then rewrite it using the duals so that the prediction is completely independent of complex feature basis Φ
3. And is further predicted using kernels

$$w^T\phi(x) = \left(\sum_n \alpha_n y_n \phi(x_n)\right)^T \phi(x) \Rightarrow \sum_n \alpha_n y_n \phi^T(x_n)\ \phi(x) \Rightarrow \sum_n \alpha_n y_n k(x_n, x)$$

4. Hence, concludes that we don't need a complex basis to model non-separable data well, and the kernel does the job.

$$d_H(\phi(x_0)) = \frac{|w^T(\phi(x_0)) + b|}{||w||_2}$$

$$w^* = arg_w max\ [min_n\ d_H(\phi(x_n))]$$

LSTM: Backpropagation through time, loss minimized through Adam or SGD, Gradients are updated through W and b accordingly

- σ: sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$
- $\tanh$: hyperbolic tangent function
- $\odot$: element-wise (Hadamard) product
- W_*, b_*: learnable weights and biases for each gate

1. Forget Gate (what to forget from the past):

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

2. Input Gate (what to store in memory):

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

3. Cell State Update:

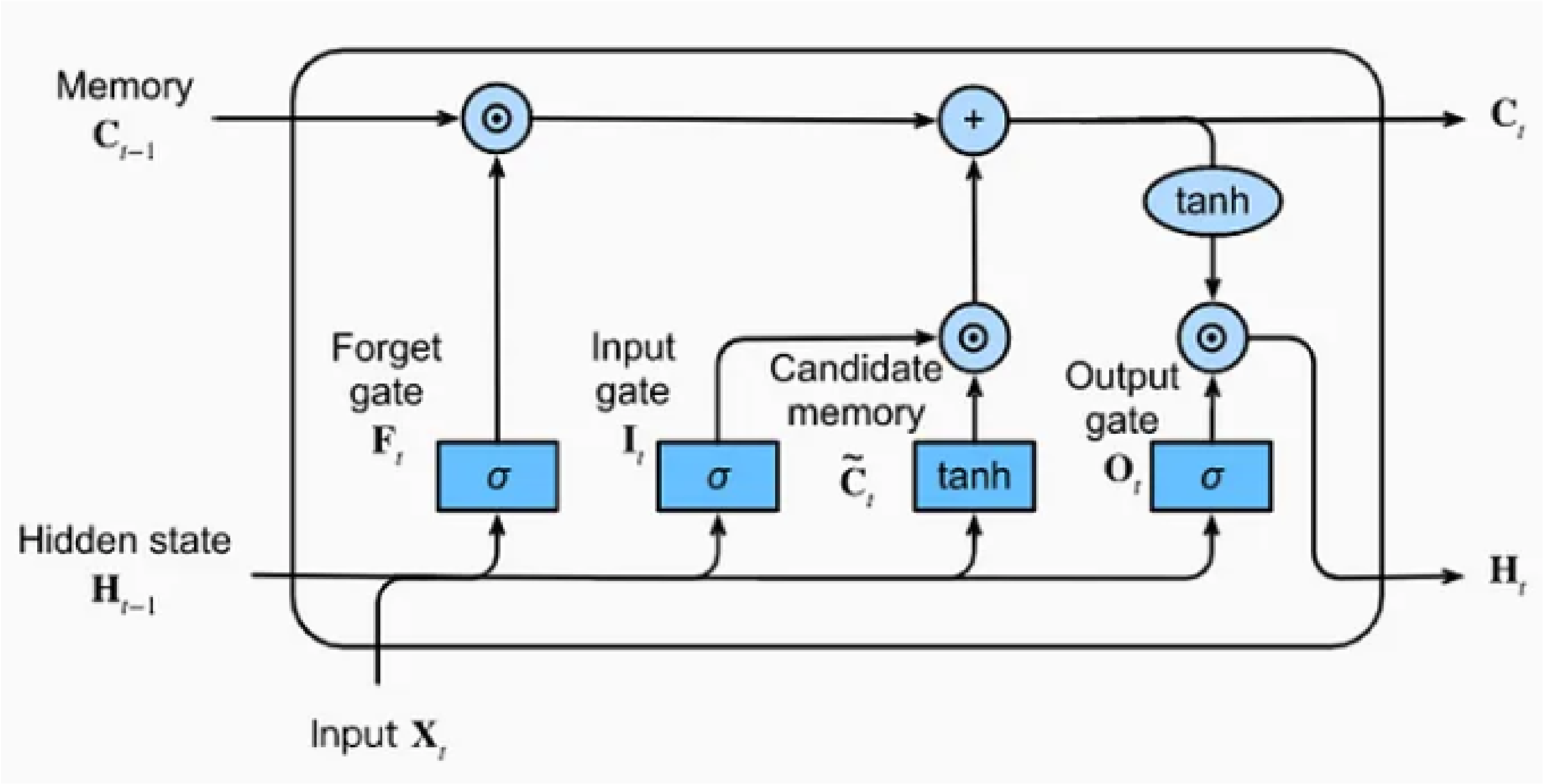
$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

4. Output Gate (what to output):

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t \odot \tanh(c_t)$$

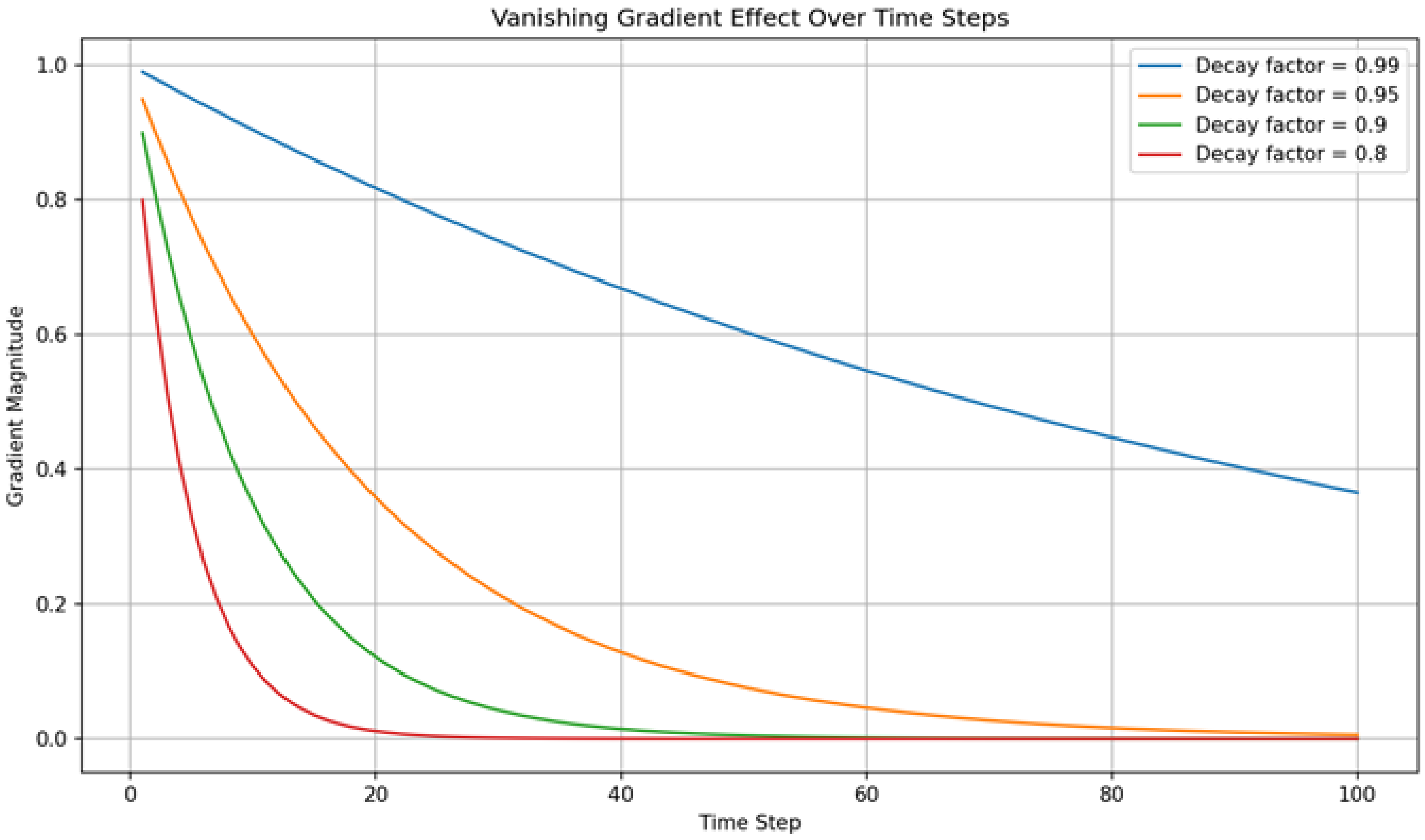
LSTM Architecture



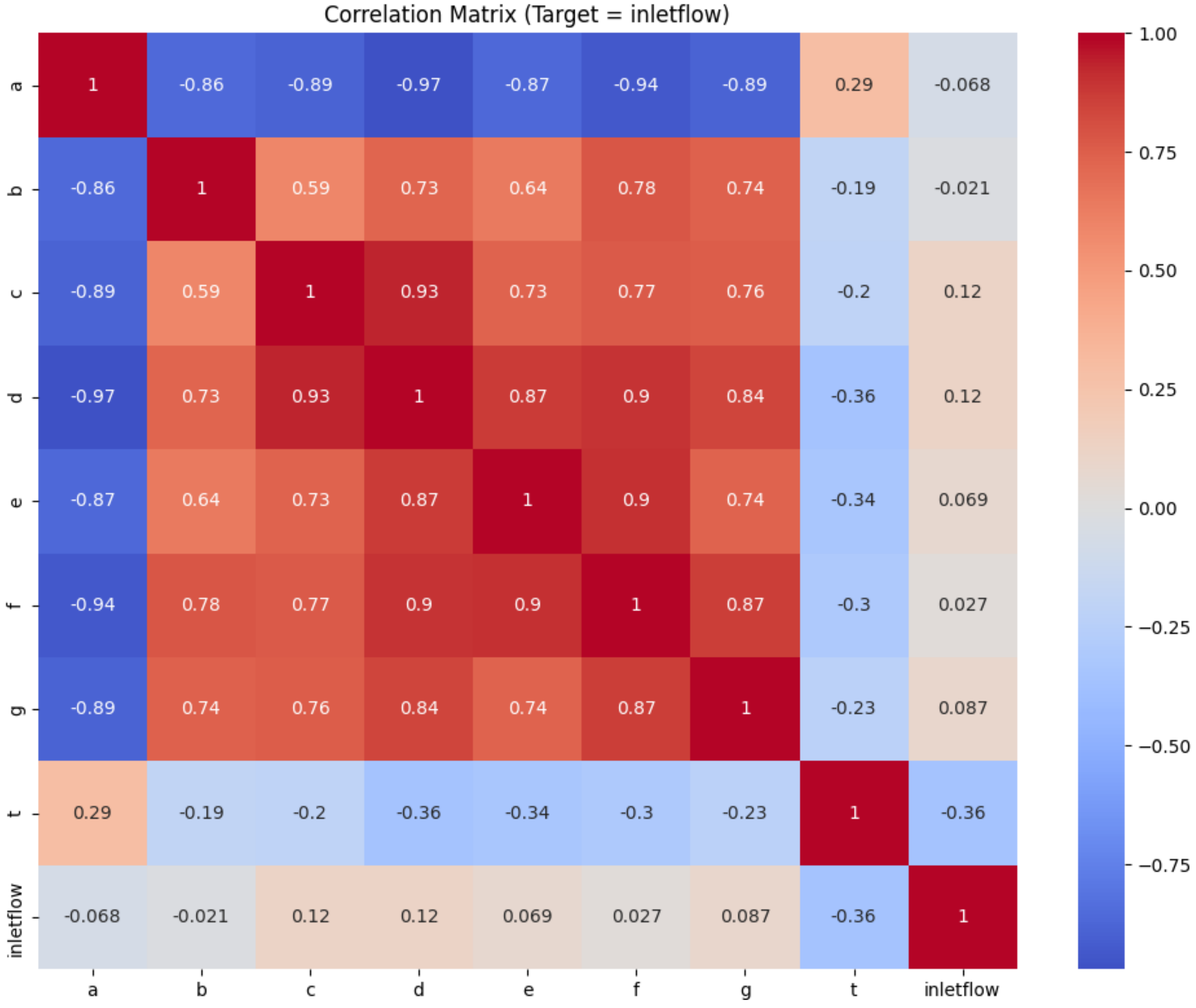
Feedforward is analogous to learning a skill.

Backward propagation: Teacher correcting the student and providing reasoning so the next time around the skill is improved.

Vanishing Gradient Problem



A note about data – Correlation or Causality




```
Entropy from a to inletflow: 1.153290676514822
Entropy from b to inletflow: 0.6132888819786857
Entropy from c to inletflow: 0.4598259267461554
Entropy from d to inletflow: 0.6829609001590522
Entropy from e to inletflow: 0.22074298069619966
Entropy from f to inletflow: 0.13632463700175002
Entropy from g to inletflow: 0.2667028999897536
Entropy from t to inletflow: 1.5797260692532442

Process finished with exit code 0
```

PM-230A – Used Cases 1

PM-230 vibration issue since 1998 commissioning, over $500,000K spent in RCA including expert input from OEM and other consulting firms. Algorithms utilized

Random Forest
- R2 Score: 0.948

XGBoost
- R2 Score: 0.939

LightGBM
- R2 Score: 0.935

K Nearest Neighbors
- R2 Score: 0.918

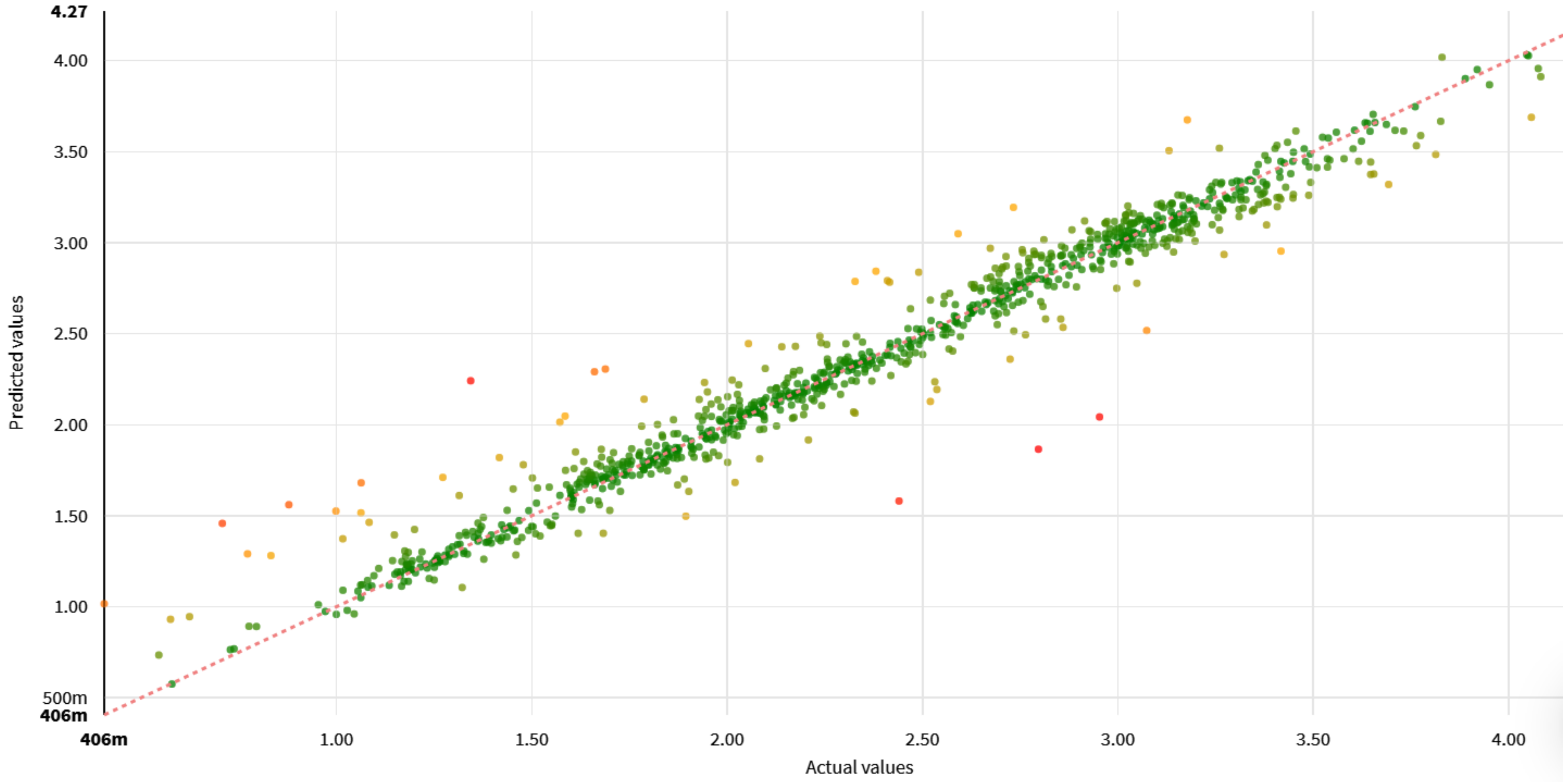
Artificial Neural Network
- R2 Score: 0.738

Ordinary Least Squares
- R2 Score: 0.639

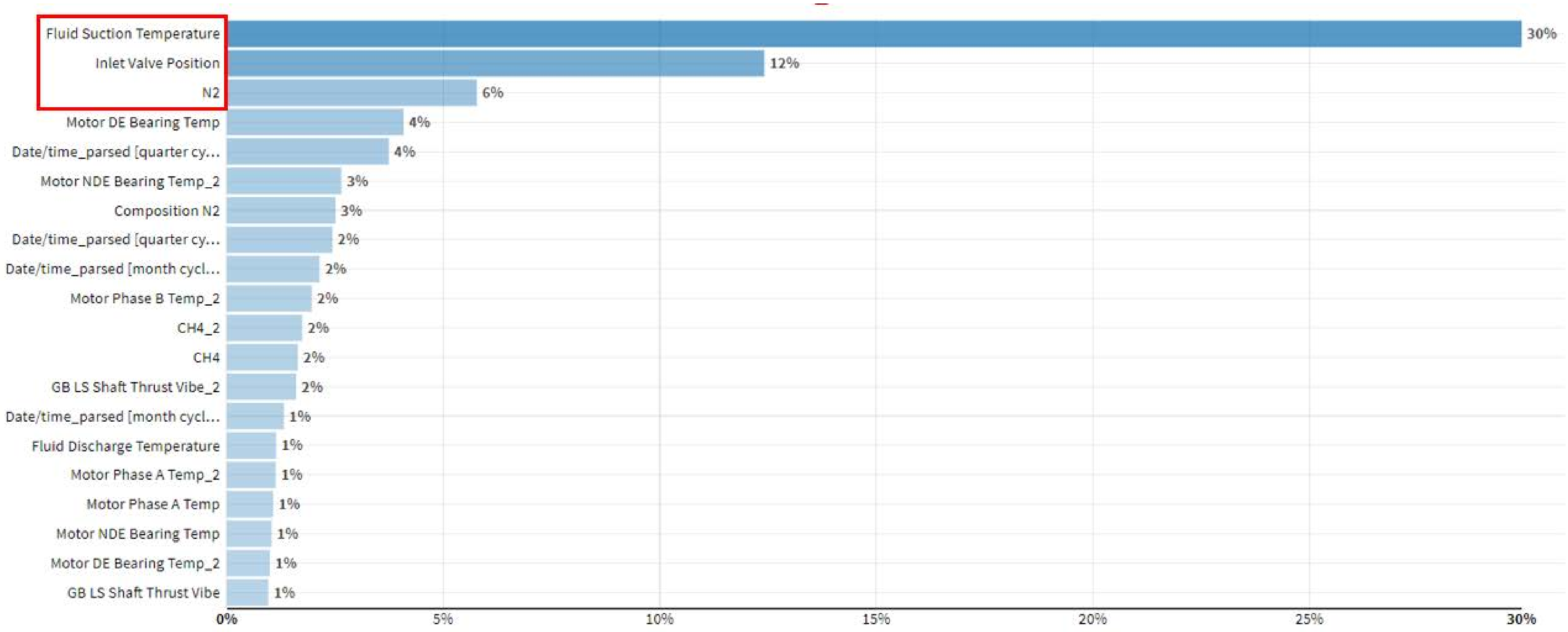
Ridge Regression
- R2 Score: 0.629

Lasso Regression
- R2 Score: 0.579

Scatter Plot



Variable Importance – Random Forest Tree Segregation



Error Metrics – Testing the Model

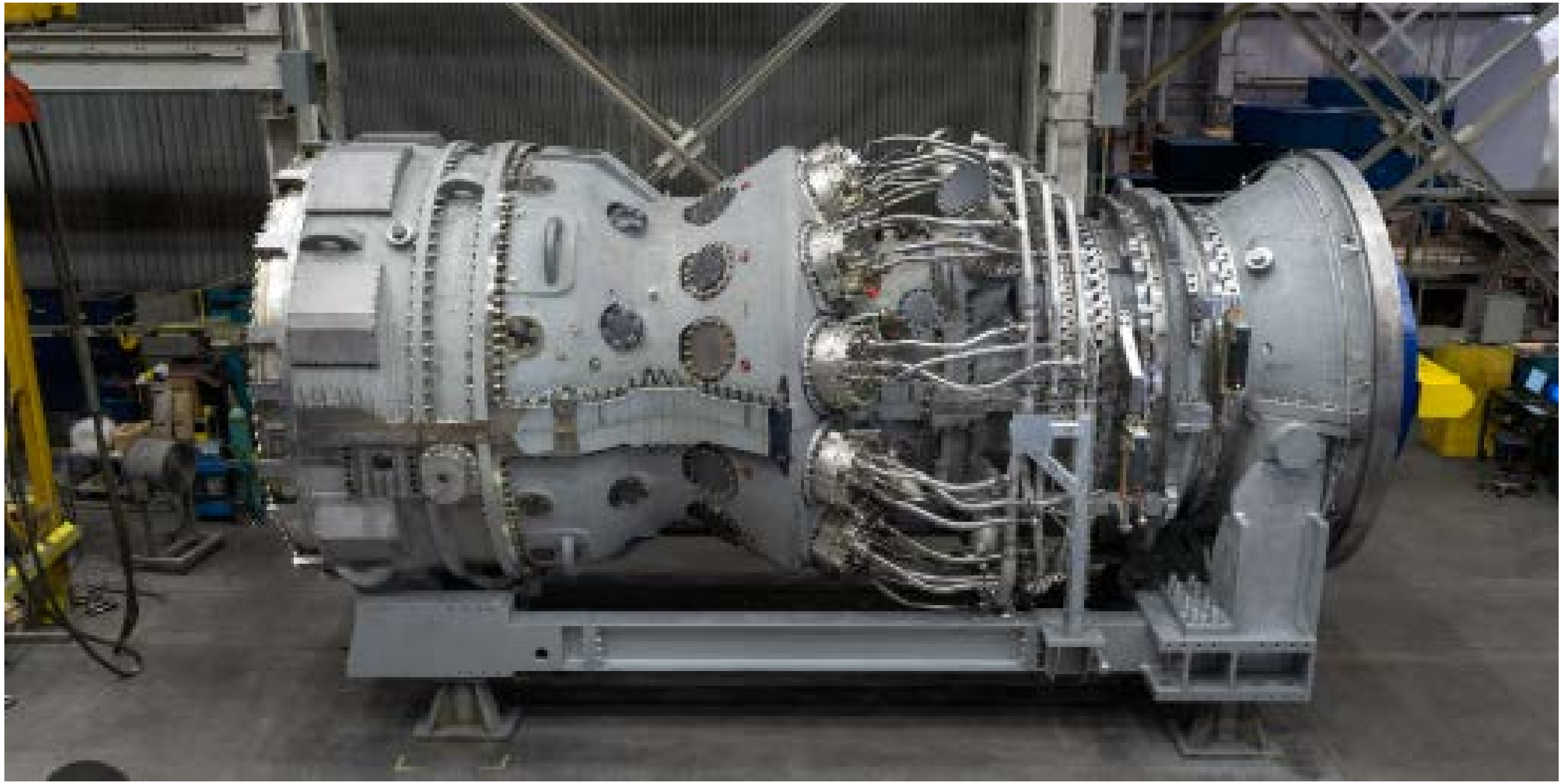
Explained Variance Score	0.94768
Mean Absolute Error (MAE)	0.088889
Mean Absolute Percentage Error	4.33%
Mean Squared Error (MSE)	0.027717

Root Mean Squared Error (RMSE)	0.16648
Root Mean Squared Logarithmic Error (RMSLE)	0.054523
Pearson coefficient	0.97358
R2 Score	0.94768

De-ethanizer Data Analytics – Case 2

Gas Turbine Case 3 – Anomaly detection (Unsupervised)

Including CNNs network to analyze lube oil images to understand oil quality and rank it based on condition



LSTM – Autoencoders, Training and Tunning

```
C:\Users\ubhachu\PycharmProjects\pythonProject\.venv\Scripts\python.exe C:\Users\ubhachu\PycharmProjects\pythonProject\ForecastingLSTM.py
2025-07-09 09:42:50.254358: I tensorflow/core/util/port.cc:113] oneDNN custom operations are on. You may see slightly different numerical results due to floating-point rou
2025-07-09 09:42:51.262174: I tensorflow/core/util/port.cc:113] oneDNN custom operations are on. You may see slightly different numerical results due to floating-point rou
=== LSTM Forecasting Model for Turbine Excursion Prediction ===

Available columns:
 ['Date', 't1', 't2', 't3', 't4', 't5', 't6', 't7']
Enter temperature sensor column names (comma-separated): t1,t2,t3,t4,t5,t6,t7
Enter timestamp column (or press Enter to skip): Date

🔎 Spread Statistics:
→ Max Spread: 347.21 °C
→ Min Spread: 26.53 °C
Enter number of lookback timesteps (e.g., 20): 20
Enter forecast horizon in steps (e.g., 5): 10
Enter spread threshold to define an excursion (e.g., 100): 34

📊 Dataset shape – X: (6848, 20, 7), y: (6848,)
📈 % of Excursions in Data: 59.01%
2025-07-09 09:43:43.012678: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instructions in performance-criti
To enable the following instructions: AVX2 AVX_VNNI FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
C:\Users\ubhachu\PycharmProjects\pythonProject\.venv\Lib\site-packages\keras\src\layers\rnn\rnn.py:204: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a
  super().__init__(**kwargs)
Epoch 1/10
155/155 ━━━━━━━━━━━━━━━━━━━━ 2s 6ms/step - accuracy: 0.4674 - loss: 0.6995 - val_accuracy: 0.1131 - val_loss: 0.7392
Epoch 2/10
155/155 ━━━━━━━━━━━━━━━━━━━━ 1s 6ms/step - accuracy: 0.4697 - loss: 0.6980 - val_accuracy: 0.1131 - val_loss: 0.7235
Epoch 3/10
155/155 ━━━━━━━━━━━━━━━━━━━━ 1s 6ms/step - accuracy: 0.4694 - loss: 0.6978 - val_accuracy: 0.1131 - val_loss: 0.7229
```

```
🔮 Predicted Excursions: 632 out of 1370 test samples

🕒 Forecasted Excursions Will Likely Occur At:
 → 4/4/2025 6:00:00
 → 4/4/2025 7:00:00
 → 4/4/2025 8:00:00
 → 4/4/2025 9:00:00
 → 4/4/2025 10:00:00
 → 4/4/2025 12:00:00
 → 4/4/2025 13:00:00
 → 4/4/2025 14:00:00
 → 4/4/2025 15:00:00
 → 4/4/2025 16:00:00
 → 4/4/2025 17:00:00
 → 4/4/2025 18:00:00
 → 4/4/2025 19:00:00
 → 4/4/2025 20:00:00
 → 4/4/2025 21:00:00
 → 4/4/2025 22:00:00
 → 4/4/2025 23:00:00
 → 4/5/2025 0:00:00
 → 4/5/2025 1:00:00
```

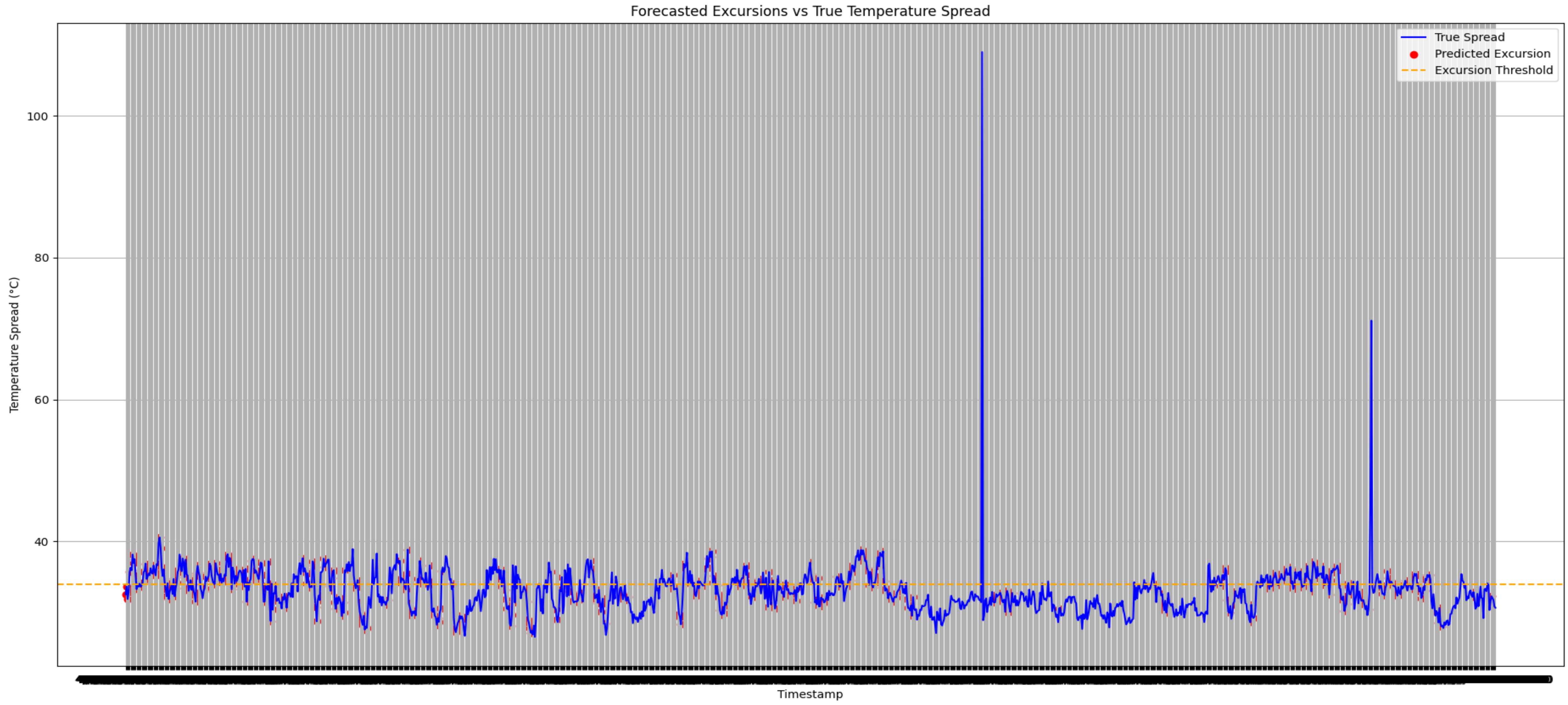
Feature Engineering, LSTM –dropout tuning

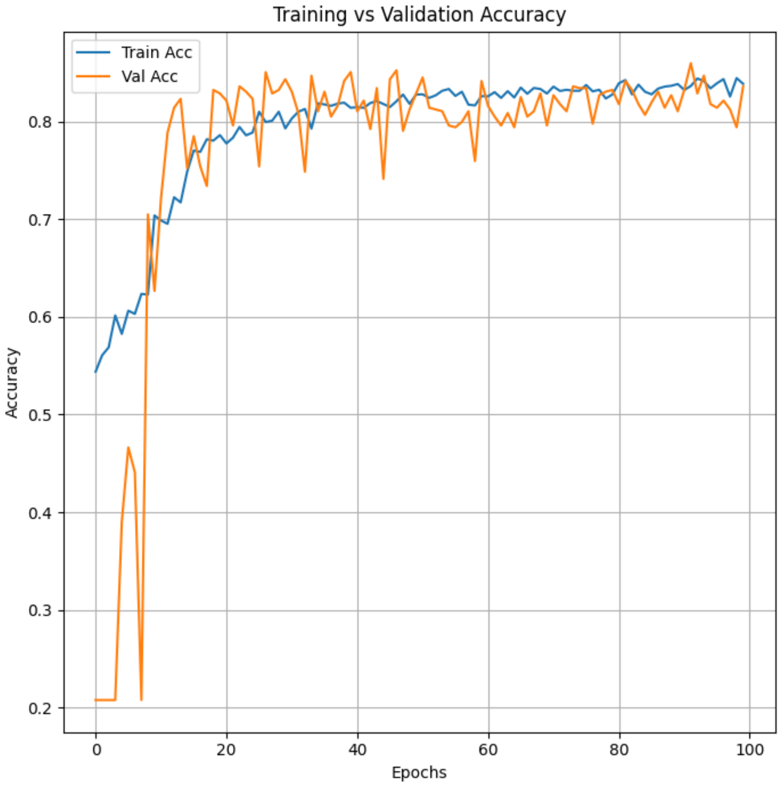
```
🔮 Predicted Excursions: 1 out of 1370 test samples

🕒 Forecasted Excursions Will Likely Occur At:
 → 6/6/2025 6:00:00

Process finished with exit code 0
```

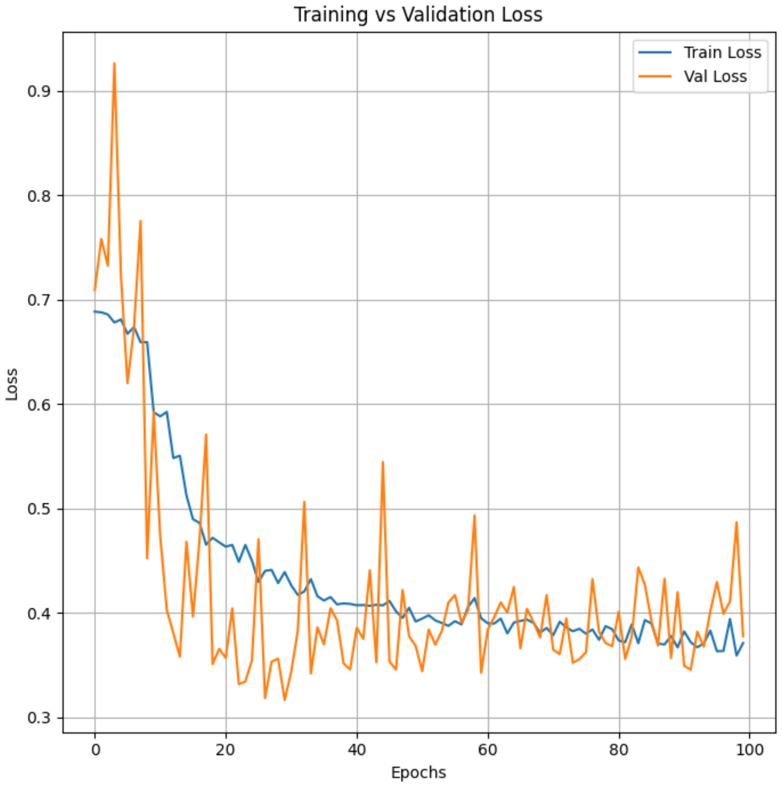
```
🔮 Predicted Excursions: 8 out of 1370 test samples

🕒 Forecasted Excursions Will Likely Occur At:
 → 6/6/2025 5:00:00
 → 6/6/2025 6:00:00
 → 6/6/2025 7:00:00
 → 6/6/2025 8:00:00
 → 6/6/2025 9:00:00
 → 6/6/2025 10:00:00
 → 6/6/2025 11:00:00
 → 6/6/2025 12:00:00
```

Epoch Change

Predicted Incursion – After EPOCH, 2 Layer LSTM dropout, Feature Engineering







Properly Trained Model – Training and validation loss both decrease smoothly, and accuracy increases similarly for both. No big gap between curves

Overfitting Model – Training loss keeps decreasing, but validation loss starts increasing after a point. Training accuracy is high, but validation accuracy drops

Underfitting Model – Both training and validation loss remain high, and accuracy stays low for both — the model has not learned the patterns well

What is the model doing and what can be changed

Observation	Interpretation	Fix
59% excursions	Good label balance	Keep as-is
1 predicted excursion	Model not learning	Increase complexity & training
Good threshold (34°C)	Within spread max	Keep for now
Failing prediction	Likely underfitting or poor features	Improve model & inputs

Model 1: Underfitting due to feature set, algorithm, epochs, training strategy. Epoch!

Metric	Observation	Interpretation
% Excursions in data	**59.01%**	Still good label balance
Predicted excursions	**20 out of 1370**	Still far too few
Validation accuracy	**~0.47**	Worse than random guessing
Loss	**~0.69**	Model stuck near untrained state

Model 2:Still not accurate further improvement with 2 Layer LSTM and dropout to avoid overfitting

Metric	Value	Interpretation
Accuracy	~0.75	Model is reliably separating classes
Predicted Excursions	715 / 1370	Balanced prediction, aligns with label distribution
Histogram	Wide, peak at 1.0	Confident predictions
Label Distribution	59% excursions	Good for binary classification

Model 2: Finally, some hope feature engineering.

y_pred = (y_pred_prob > 0.5).astype(int) more confidence change to 0.7, 0.8 and carry on……

Voice of Industry

Purposeful AI

- AI implemented with intent where applicable
- Leveraging of Generative AI to empower users

Digital Integrated Insights And Workflows

- Modularity empowered by integrations and interactions.
- Digital and simplified work processes

Data Availability

- Democratized data and insights for all users
- Enablement of decision making with data

Semi or Automated Operations

- Remotely monitored operations (digital twin supported)
- High resolution data enabling process/operation optimization

Prescriptive Analytics And Early Prevention

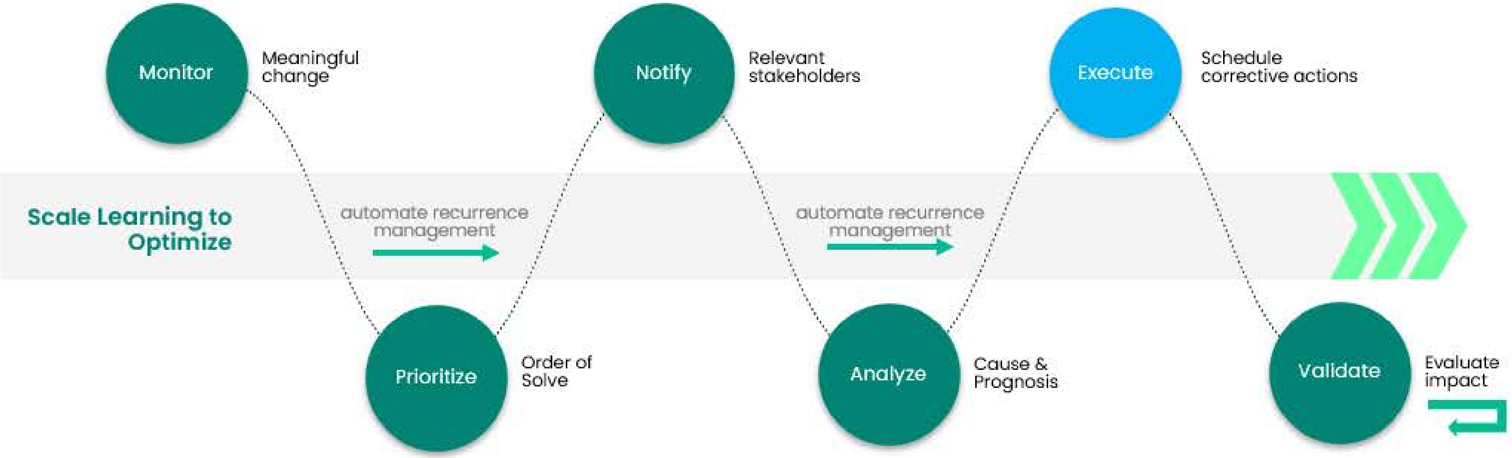
- EaAI or Physics driven detection of anomalies potential failure
- Risk prioritization and failure mode identification

Change Management Supporting Digital Roadmaps

- Recognition of the implications of a digital shift to the workforce
- User work process definition and task enablement

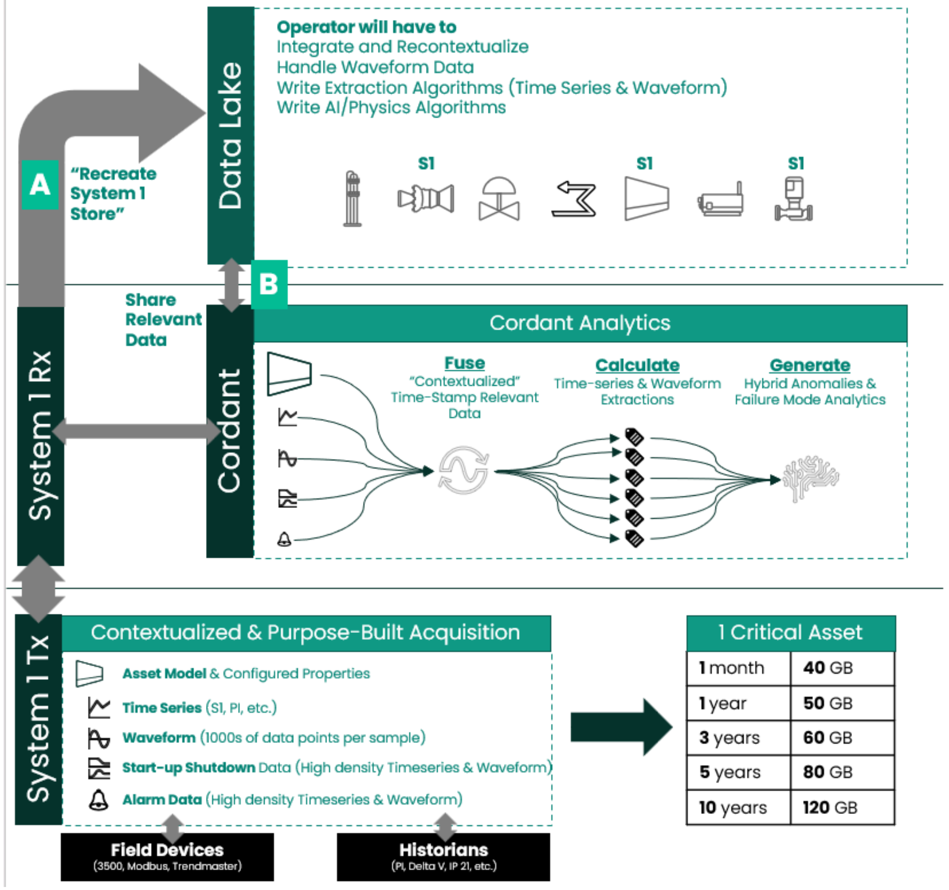
Holistic asset health management

System-driven, consistent and collaborative work practice with analytics supporting continuous improvement



3rd Dimension for Integrated Cause & Effect

Constituent Capabilities



1 Critical Asset	
1 month	40 GB
1 year	50 GB
3 years	60 GB
5 years	80 GB
10 years	120 GB

Baker Hughes Goal

Focus on Core Competencies
Implementation Efficiency
Minimize Redundant Data Storage
Integrate and Enable Other Technologies

A: Import S1 source data Into Data Lake

- Can be done with or without Cordant, However latter has limitations
- Heavy lift for customer to integrate multiple source S1, duplicate entire Db, build asset models, remap all source tags, create waveform extractions, create/deploy/sustain AD and FM analytics etc.

B: Share Load Between Cordant & Data Lake

- Have Cordant natively ingest and process heavy mechanical and process algorithms and integrate outputs to lake/other as needed
- Leverage lake and other analytics for application use cases beyond Cordant purview with integrated results ingestion and fleet management workflow via Cordant

A/B: Enable a mix

- Identify use case and business benefit to explore leveraging both methods for research and expansion

Baker Hughes

Advantages

Contextual Intelligence (Asset Models – 60+ yrs, inter-dependencies studies.
Consistency Across Platforms
Advanced Analytics
Scalability
Cross – Asset Cause and Effect Analysis
Preferential data, Footprint on rotating assets, system 1 data integration.
Thermodynamics critical calcs.

Risks

Data sharing by organizations will impact model accuracy
Flexibility for Non-Standard Assets
Complexity in Initial Setup
Accurate MetaData
Potential integration with external systems

Benefits

- Decision making is made more data driven vs anecdotal. Including RCA process to save costs by assessing issues around repeat failures.
- Business advantages in saving costs during various stages of development and in different disciplines. Many application to name.
- Optimization of process and cost value benefits driving decision make to increase production, reduce bottlenecks and become competitive.
- Mechanical engineering can benefit from improved diagnostics, data assessment and failure analysis as part of the reliability program.
- Applications in emissions and energy optimization (optimal running conditions for equipment for lower emissions, data correlation, trends) .
- Safety incident predictions (https://www.sciencedirect.com/science/article/pii/S0098135422001272) and risk-based assessment.

WHAT is Required and WHAT it is not?

- Data assessment and analysis before machine learning and AI- Good understanding of statistics/mathematics to understand data and analyze it and figure out what it saying.
- Understanding Algorithm structure and what they are in the Machine learning/AI world and what their respective advantages/disadvantages. Using GRU (less gated architecture, more stable gradients) vs LSTM (more gates in architecture, relatively less stable) in achieving time series prediction. Both can achieve the same thing but different outcomes, over vs underfitting, training set and time, hardware requirements.

Team of experienced programmers, data analysts and SMEs

- Data Collection, Data preparation, Choosing a type of model, training the model, evaluating the model
- Turning of parameters, making predictions and optimizing and saving cost .

References

https://medium.com/@ottaviocalzone/an-intuitive-explanation-of-lstm-a035eb6ab42c

https://www.gevernova.com/gas-power/resources/

https://medium.com/@venunagender2111/what-is-machine-learning-0583ca054e52

https://www.ddhk.org/?d=277708916

https://shreemsolutions.ca/

Questions