ISYE7406 Project Group 2 Final Report

Predicting Wind Turbines Defaults

Feng Gao 903533870

Leiru Long 903940713

Xinhao Zhou 903949668

Yuhan Fu 903946247

Ziyun Wang 903880511

SCHOOL OF INDUSTRIAL AND SYSTEMS ENGINEERING

April 24, 2024

Table of Contents

1 Executive Summary

In our project, we focused on improving the predictive accuracy of wind turbine defaults through the use of advanced data mining techniques and a diverse array of machine learning models. Our approach was based on thorough data preprocessing and feature engineering, complemented by the deployment of several models, including Isolation Tree, Support Vector Machine (SVM), Vanilla Neural Network, and Long Short-Term Memory (LSTM).

The preparation of the turbine dataset for predictive modeling involved meticulous data cleaning and preprocessing to enhance model accuracy and efficiency. Key methods included backward fill for addressing missing data, which maintains temporal continuity without the need for future data; transforming data frequency from 10-minute intervals to hourly to reduce noise; and handling multicollinearity by identifying and excluding highly correlated variables. Additionally, data smoothing techniques and feature scaling were applied to expose underlying trends and ensure uniform feature influence. To tackle the challenge of data imbalance, the Synthetic Minority Oversampling Technique (SMOTE) was employed, alongside expanding the default classification with a 60-day predictive window to better detect rare failure events. The dataset was further validated using a cross-validation strategy that reserved 20% of the data as a validation set, ensuring that the model remains effective and generalizable to new data.

We implemented several machine learning methods to enhance the prediction of turbine faults, each demonstrating unique capabilities and facing distinct challenges.

- **Isolation Forest (IF):** chosen for its suitability in datasets with few failure instances, uses isolation trees to detect anomalies but struggled with precision.
- **Support Vector Machine (SVM):** utilizes an RBF kernel and adjusted class weights to reduce false positives, showed improved accuracy but faced challenges with low precision and recall.
- **Vanilla Neural Network (NN):** designed with various mechanisms to prevent overfitting and incorporating a variety of temporal features, performed well in specific systems but exhibited tendencies to overfit to limited default data.
- **Long Short-Term Memory (LSTM):** ideal for sequential data and equipped with advanced gating mechanisms, also underperformed due to low recall and precision and incurred negative savings, which was partly due to overfitting

Our exploration of different predictive models revealed varied performances. Specifically, the Vanilla Neural Network proved to be particularly effective, achieving an accuracy of 73.792%, a precision of 44.792%, and a recall of 32.242% in the hydraulic subsystem, demonstrating a promising balance of accuracy and utility. This model showcased its potential for reducing maintenance costs and minimizing turbine downtime, highlighting the economic benefits of predictive maintenance.

In the final evaluation on the test dataset consisting of unseen data from 2017-09-01 to 2017-12-31, our Vanilla Neural Network achieved **a total savings of €14,687.04**, further underscoring its practical value in operational settings. This performance emphasizes the significant cost advantages that can be realized through the application of sophisticated predictive models in the maintenance of wind turbines.

2 Wind Turbines Default Data

2.1 Data Insights and Analysis

In this analysis, we delve into data insights that enhance our understanding of the variables influencing wind turbine operations. The turbine comprises five subsystems: Generator, Generator Bearings, Gearbox, Transformer, and Hydraulic System. A total of five turbines are studied. Wind turbines are subject to a myriad of complex factors during operation, including environmental conditions such as changes in wind speed and ambient temperature, as well as the temperatures of the equipment itself. These factors may induce alterations in critical components, potentially impacting the equipment's health and longevity. Accordingly, it is prudent to predict future failures by analyzing trends in environmental changes and other relevant data. Initially, each subsystem is examined to glean insights.

Focusing on the generator bearings, it is observed that the temperatures at the Non-Drive End and Drive End may serve as indicators of bearing load and wear. Fluctuations in these temperatures might suggest impending bearing failure. Additionally, ambient temperatures significantly influence the bearings' heat dissipation capabilities, directly affecting their operation and lifespan.

Data for the sign variables of the Non-Drive End and Drive End bearings were plotted over various intervals: one week, one, two, three, and six months, as well as one year and one and a half years prior to failure. The resulting plots, Figures 2-1 and 2-2, illustrate the temperature distribution for the Non-Drive End and Drive End bearings, respectively. The data for the Non-Drive End bearing shows a broad distribution over time without a clear upward or downward trend in median temperature change, particularly in the months leading up to failure. Similarly, the temperature distribution for the Drive End bearing does not exhibit significant fluctuations, nor a consistent rise or fall in temperature as failure approaches. These findings suggest that monitoring temperature alone may not be adequate for predicting bearing failure.

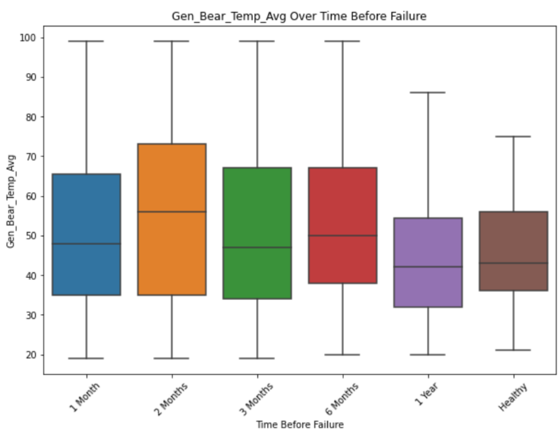

Figure 2-1 Temperature Distribution of the Non-Drive End Bearing

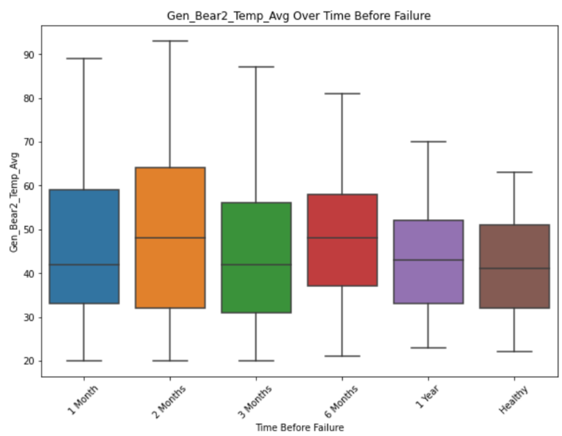

Figure 2-2 Temperature Distribution of the Drive End Bearing

Moreover, the difference between ambient and bearing temperatures, depicted in Figure 2-3, shows that the median change from one month to one year is not significant, indicating that the bearing temperature remains relatively stable relative to the ambient temperature during these intervals. The similarity in data distribution between the "healthy" state and the "one year" period prior to failure suggests that the relative influence of environmental conditions on bearing temperature may not have undergone significant changes.

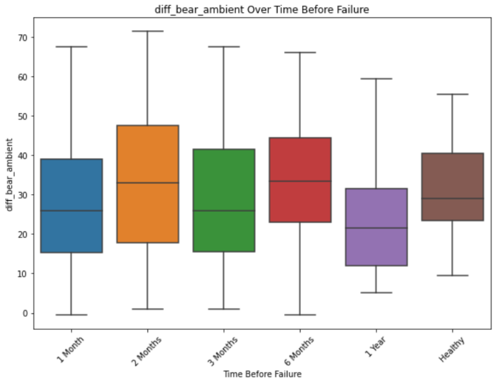

Figure 2-3 Distribution of Temperature Difference Between Bear and Ambient

For the other four subsystems, similar analyses were conducted. Most of these results displayed a wavy pattern without a clear trend, complicating the predictive analysis.

Additionally, data across different turbine IDs throughout the training period were also plotted, revealing various scenarios as illustrated in Figures 2-4 and 2-5. Some turbines exhibited outliers for certain variables, or a distribution of variables that differed from other turbines. This variability suggests the potential application of methods such as moving deviation in subsequent data preprocessing.

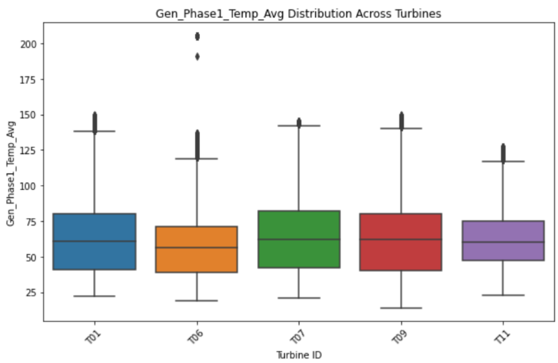

Figure 2-4 Distribution across Turbines for GenPhase

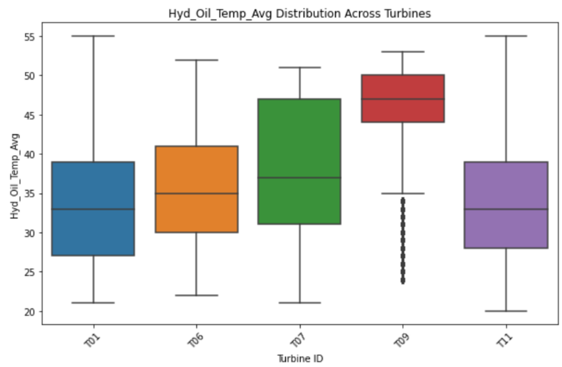

Figure 2-5 Distribution across Turbines for HydTemp

2.2 Data Cleaning and Pre-processing

The following sections outline the key steps taken to prepare the turbine dataset for modeling. Proper data cleaning and preprocessing are essential to enhance the accuracy and efficiency of the predictive models aimed at forecasting turbine failures.

By refining the dataset, we create a robust foundation for model training. This processed data helps us more reliably predict potential turbine failures, enabling more precise and timely maintenance interventions.

Data Imputation with Backward Fill

To address missing data within the turbine dataset, we employed the backward fill method, which replaces missing values with the most recently observed data prior to the gap. This approach helps maintain the temporal continuity of our time series data, preserving the inherent correlations and trends crucial for our analysis. Forward filling, although very popular, is unsuitable for our model because it relies on future data, which isn't accessible in real-time operations. Backward filling allows us to impute missing data using only historical information, aligning with practical constraints.

Frequency Transformation

Originally captured at 10-minute intervals, the dataset was transformed to an hourly frequency to reduce noise and enhance the detectability of significant trends. This transformation simplifies the dataset, making it computationally efficient while retaining essential dynamics pertinent to turbine operation.

Handling Multicollinearity

The presence of multicollinearity increases the variance of the coefficient estimates, making subsequent modeling more complex. As shown in Figure 2-6, in our complete set of features, many pairs are highly correlated. To address this issue, we set a correlation threshold of 0.8, on the basis of which we identified pairs of variables with absolute correlation above the threshold, indicating strong correlation. Following which, we systematically excluded one variable from each pair of highly correlated features, selecting the variable to keep based on alphabetical precedence, thus minimizing redundancy without losing a lot of information. Figure 2-7 illustrates the correlations of the features after removing multicollinearity. The set of features kept after filtering is as listed in Table 2-5.

Table 2-5 Selected Features as Input Data to Our Models

Gen_RPM_Std	Hyd_Oil_Temp_Avg	Gen_Bear2_Temp_Avg	HVTrafo_Phase1_Temp_Avg
Gear_Bear_Temp_Avg	Var_Windspeed1	Avg_Winddirection2	Var_Winddirection2
Avg_AmbientTemp	Avg_Pressure	Avg_Humidity	Avg_Precipitation
Anemometer1_Avg_Freq			

Data Smoothing Techniques

Taking into considerations the time-series nature of our data, we also applied both moving averages and moving standard deviations to the turbine dataset. This smoothing technique not only reveals underlying trends in the data, but also dampens fluctuations and captures more persistent patterns that are critical to correctly predicting failures.

Feature Scaling

To ensure that each feature contributes equally to the model when it is trained, and to prevent any single variable from dominating the model because of its size, we normalize the features to have zero mean and unit variance. This standardization aligns the data distribution closer to a normal distribution, facilitating more effective learning by the predictive models.

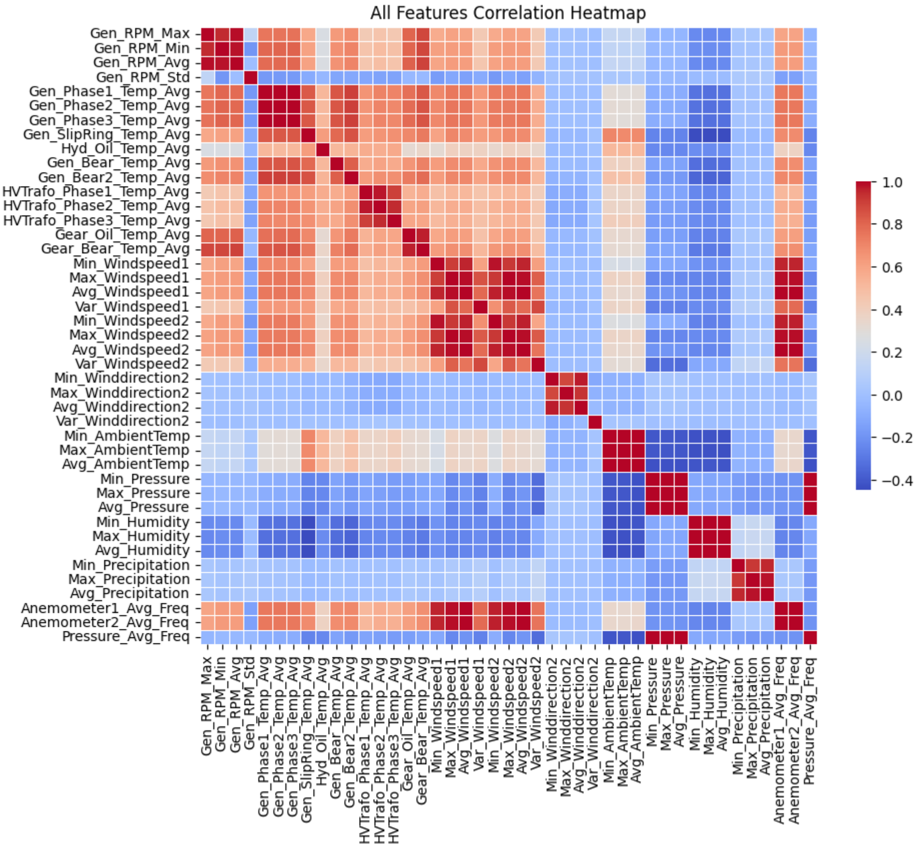

Figure 2-6 Correlation Heatmap of Features with Non-zero Variances

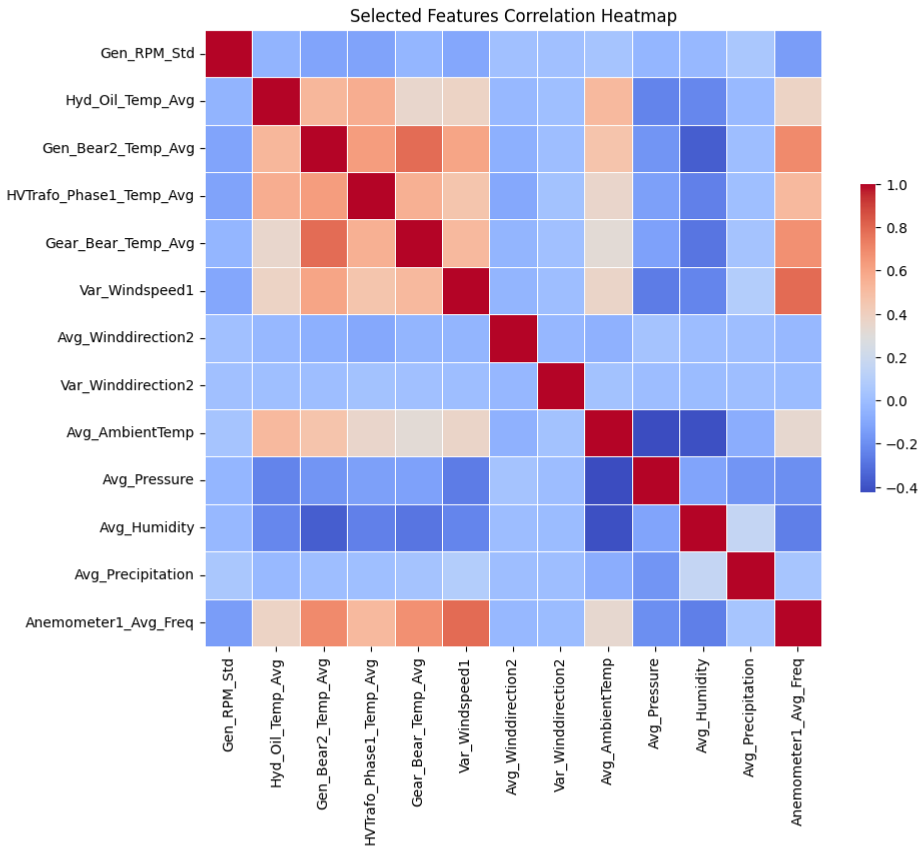

Figure 2-7 Correlation Heatmap of Selected Features with No Multicollearity

Addressing Data Imbalance

In our turbine dataset, turbines default events are much less frequently than normal events, thus making our dataset extremely imbalanced. To improve the machine learning model's ability to detect the few events, we expanded the default classification by labeling a 60-day window before a failure as an impending failure. Although the turbine may not experience problems during the early part of that window of operation and thus may introduce some noise, we consider this noise acceptable for the potential benefit of identifying failure precursors.

To address the imbalance between default and non-default events further, we employ the Synthetic Minority Oversampling Technique (SMOTE). This method artificially generates new examples from the minority class, enriching the dataset and helping the model learn to detect rare failure events more effectively.

Cross Validation

We retain the last 20% of the time series in the training set as a validation set in order to more effectively validate the effectiveness of our model and ensure that the model can be generalized to completely new data. This approach is part of our cross-validation strategy, which allows us to build the model's predictions and adjust the parameters to make them as good as possible before actually applying them to the test data.

3 Methodology

3.1 Isolation Forest

To address the challenge of predicting turbine failures, we employ the Isolation Forest model, a choice influenced by the limited number of failure examples in our dataset. The dataset comprises 68,571 hourly data points, yet features only 23 recorded instances of turbine failures. This imbalance renders the task predominantly an outlier detection challenge.

The Isolation Forest stands out as an unsupervised learning algorithm specifically designed to identify anomalies. It diverges from traditional methods that rely on distance or density metrics. Instead, it randomly selects a feature and a corresponding split value between the feature's maximum and minimum values. By constructing multiple Isolation Trees to create a forest, the model increases the probability of promptly isolating outliers. The path length in these trees, from the root to the leaf, serves as an indicator of normality; shorter paths typically signify an anomaly, thereby effectively pinpointing unusual data points that could indicate turbine failures.

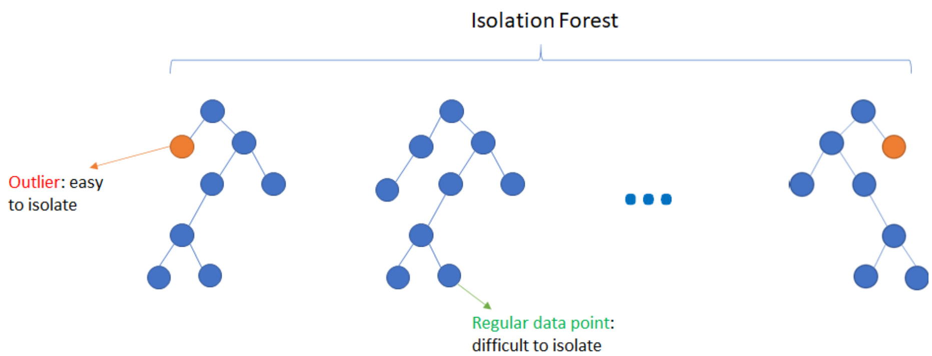

Figure 3-8 Structure of Isolation Forest

The figures below illustrate the results of the Isolation Forest model on the hydraulic system. The blue vertical lines denote the outliers classified by the model, while the area surrounded by orange dashed lines represents the expected region where true label anomalies should fall. However, as observable, the model does not effectively capture the failure instances. It achieved a precision rate of only 9.2% in identifying defects within the hydraulic group.

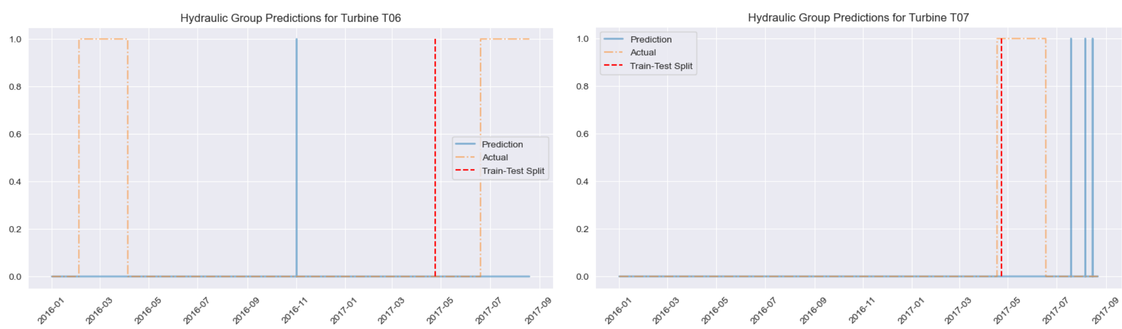

Figure 3-9 Demonstration of Isolation Forest Performance

Upon further analysis, it is evident that while our feature selection was robust, there is substantial room for improvement in the performance of the Isolation Forest model through enhanced feature engineering. Observations indicate that the suspected outliers do not significantly deviate in magnitude from the normal data range when examined through individual features. This suggests that a more effective differentiation of outliers may require processing features to highlight their cumulative upward and downward trends around the failure data points. Techniques such as cumulative sum (CUSUM) could be advantageous in this context. Unfortunately, due to time constraints, we were unable to demonstrate this technique.

3.2 SVM

In this section, we discuss the application of the Support Vector Machine (SVM) algorithm to predict turbine faults in our dataset. Given the standard scaling of our data, the Radial Basis Function (RBF) kernel was chosen as it best fits the characteristics of our data.

The primary challenge in predicting turbine defaults is minimizing false positives, which can lead to unnecessary and costly maintenance operations. To address this, we optimized our SVM model by adjusting the class_weight parameter. Specifically, we set the weights to 5 for the non-default class and 1 for the default class. This adjustment significantly increases the penalty for misclassifying the non-default class, thereby reducing false alarms and enhancing the overall predictive accuracy of our model.

Figure 3-10 illustrates the SVM model's predictions in the gearbox subsystem. The orange dotted lines represent the actual turbine faults within a 60-day prediction window, highlighting the model's effectiveness in forecasting these events.

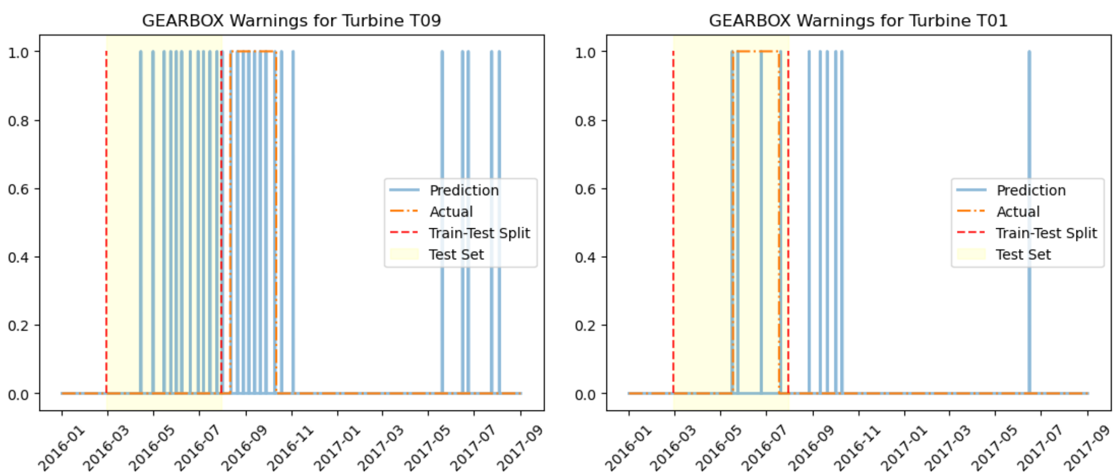

Figure 3-10 Demonstration of Performance of Vanilla Neural Network

Besides, the red dotted line marks the data split between the training and test sets. This split is not a typical random or chronological division but is structured to ensure that our training set includes at least one instance of a turbine fault, and our test set contains a complete fault window. This methodological choice allows our model to learn from a full event cycle, which is essential for recognizing the patterns that lead to a turbine fault.

Despite these strengths, the model does exhibit signs of overfitting, particularly evident in the false alarms for turbine T09. This suggests that our feature engineering might not have fully captured the complexities of the time series data, necessitating further refinement.

Anomaly Detection with One-Class SVM

Additionally, we also experimented with a one-class SVM for anomaly detection. One-Class SVM is a specialized version of the Support Vector Machine algorithm designed for anomaly detection, where the goal is to identify rare events or observations that differ significantly from the majority. A critical parameter in One-Class SVM is ν, which controls the trade-off between the fraction of outliers (anomalies) and the number of support vectors used by the model. Essentially, ν represents an upper bound on the fraction of training errors and a lower bound on the fraction of support vectors. For instance, setting $\nu = 0.05$ ensures that no more than 5% of the training examples are misclassified as outliers. Here We set ν equal to the ratio of the data points labeled as default (including the 60-day window prior to a default event) to the total number of observations in our unbalanced training set.

Despite the theoretical appeal of One-Class SVM for our application, the practical outcomes were less than satisfactory. The confusion matrix and performance metrics in Table 3-10 illustrate the model's performance on the Gearbox validation set:

Table 3-10 Confusion Matrix of One-Class SVM on Gearbox Validation Set

	Predicted Non-default	**Predicted Default**
Actual Non-default	46312	1253
Actual Default	1262	12

The performance metrics derived from the confusion matrix are in Table 3-11:

Table 3-11 Performance Metrics of One-Class SVM on Gearbox Validation Set

	Precision	**Recall**	**F1-Score**	**Support**
Class 0 (Non-default)	0.97	0.97	0.97	47565
Class 1 (Default)	0.01	0.01	0.01	1274
Accuracy			0.95	48839
Macro Avg	0.49	0.49	0.49	48839
Weighted Avg	0.95	0.95	0.95	48839

The metrics indicate a high rate of correctly predicting non-default cases but a poor performance in detecting actual defaults, with both precision and recall for defaults at only 1%. It shows that the One-Class SVM model fails to serve our primary objective of effectively predicting turbine faults.

3.3 Vanilla Neural Network

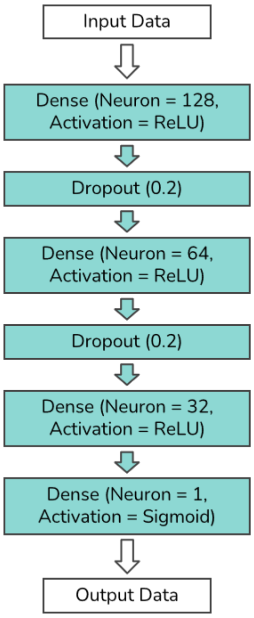

Figure 3-11 Structure of Vanilla Neural Network

As shown in Figure 3-11, our vanilla neural network comprises three dense layers, each designed to capture different levels of abstraction from the input data. To combat the prevalent challenge of overfitting, we interlace these layers with two dropout layers, each dropping out 20% of the neuron activations. Early stopping is another crucial component of our training process; it halts the training when there is no further improvement in validation loss, thus preventing the model from learning noise and irrelevant patterns.

The input data fed into the network includes hourly features, along with 1-day, 3-day, 1-week, and 2-week feature averages and variances. We also incorporate the rates of change over day-to-day, 3-day-to-3-day, week-to-week, and 2-week-to-2-week periods. This ensemble of features is intended to provide the model with a blend of immediate and extended temporal insights, enabling it to discern both short-term anomalies and long-term trends that may indicate an impending failure.

To reduce the frequency of warning and prevent consecutive warning, we implement a filtering mechanism. The model only issues a new warning if there has been a clear pause of 7 days without alerts. This approach aims to provide a buffer against premature or unnecessary warnings, thereby enhancing the reliability of our predictions.

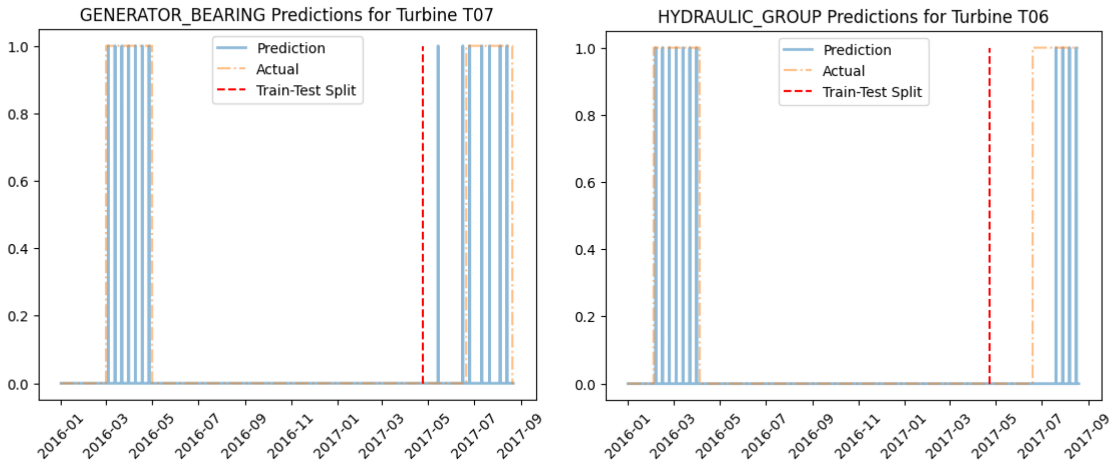

Figure 3-12 Demonstration of Performance of Vanilla Neural Network

The performance of our model, as depicted in the examples shown in Figure 3-12, demonstrates a capacity for effective default prediction within the established timeframe. The orange dotted line marks the actual turbine defaults and the 60-day window that precedes them, while the blue lines are warnings issued by our model. The red dotted line splits the timeframe into training and validation periods, where the right side of the red dotted line includes all data unseen by the model.

For a comprehensive analysis of the model's performance, we have included a detailed report in Table 3-12. This table provides evaluation metrics on 20% validation set, as well as a calculation of cost savings achieved over the entire dataset period. The model showed high accuracy in the Transformer system at 89.05%, but Gearbox and Generator systems recorded 0% for both precision and recall due to no positive predictions in the validation set. Despite these discrepancies, the model demonstrated considerable economic value, with cost savings notably high for the Gearbox at 118,333.33 €. The results suggest that while the model's predictive capabilities varies across different systems, it is economically beneficial, achieving positive savings for all systems.

Table 3-12 Performance of Vanilla Neural Network on Turbines and Systems

System	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)	Savings (€)
Transformer	89.050	58.844	35.619	44.377	97650.00
Gearbox	86.630	0	0	0	118333.33
Hydraulic Group	73.792	44.792	32.242	37.495	64233.33
Generator Bearing	83.146	38.823	63.194	48.097	71583.33
Generator	83.060	0	0	0	77750.00

However, there is an observable tendency for early warning issuance. Specifically, it tends to issue warnings too early in the 60-day window, where the turbine may still be operating normally. This inclination towards early warnings may indicate a degree of overfitting to the training data. Despite the safeguards against overfitting that we have integrated, the model still seems to be excessively influenced by the limited default instances it was exposed to during training. Given the infrequency of defaults, our model appears to have learned too intensively from the sparse default data, which hampers its ability to generalize to new, unseen data.

3.4 Long Short-Term Memory (LSTM)

In addition to vanilla neural networks, which primarily excel in handling straightforward patterns, we also explored the deployment of more sophisticated deep learning models to enhance our predictive accuracy. Among these, the Long Short-Term Memory (LSTM) model stands out due to its unique capability to process sequences and its robustness in dealing with complex time-series data.

The decision to extend our analysis to LSTM models was driven by their architectural advantages over manipulation of time series dataset. Each unit in an LSTM model is equipped with three gates: the input gate, the forget gate, and the output gate. These gates collectively ensure that the model can selectively remember, forget and process information, which is crucial for maintaining relevant historical data over time. This functionality is particularly beneficial for our application in wind turbine fault prediction.

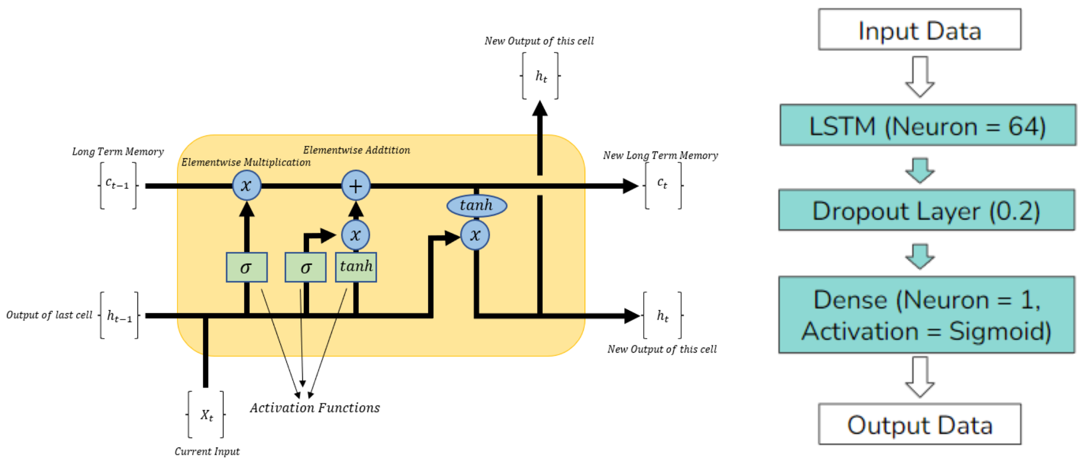

Figure 3-13 Structural Methodology of LSTM

For our specific case, we structured our LSTM network with 64 neurons, incorporated a dropout layer with a rate of 0.2 to mitigate the risk of overfitting, and concluded with a dense layer using a sigmoid activation function. We utilized a time step of 144 (equivalent to 6 days of data) to capture the temporal dependencies relevant to predicting turbine faults. Furthermore, to optimize our model, we conducted a grid search across various epochs and batch sizes, allowing us to pinpoint the most effective model configuration for our predictive needs.

Despite achieving a relatively high accuracy of over 80%, the results of the LSTM model are not satisfactory. Both the recall and precision metrics are considerably low, indicating a substantial number of false positives. This issue is clearly demonstrated in the signal visualization graph and the calculated savings, where the model's limitations become apparent.

Table 3-13 Performance of LSTM (Hydraulic Group)

Metrics	Value
False Positive	11808
False Negative	0
Total Savings	-55019444

Throughout our experiments, we observed significant fluctuations in performance metrics. There were instances where the model exhibited low accuracy but high recall, and vice versa. This inconsistency made it challenging to achieve a balanced trade-off between these metrics. After testing various configurations, including multiple LSTM layers and different neuron counts, we ultimately found that a single-layer LSTM provided the most stable results. Unfortunately, even this configuration did not meet our expectations.

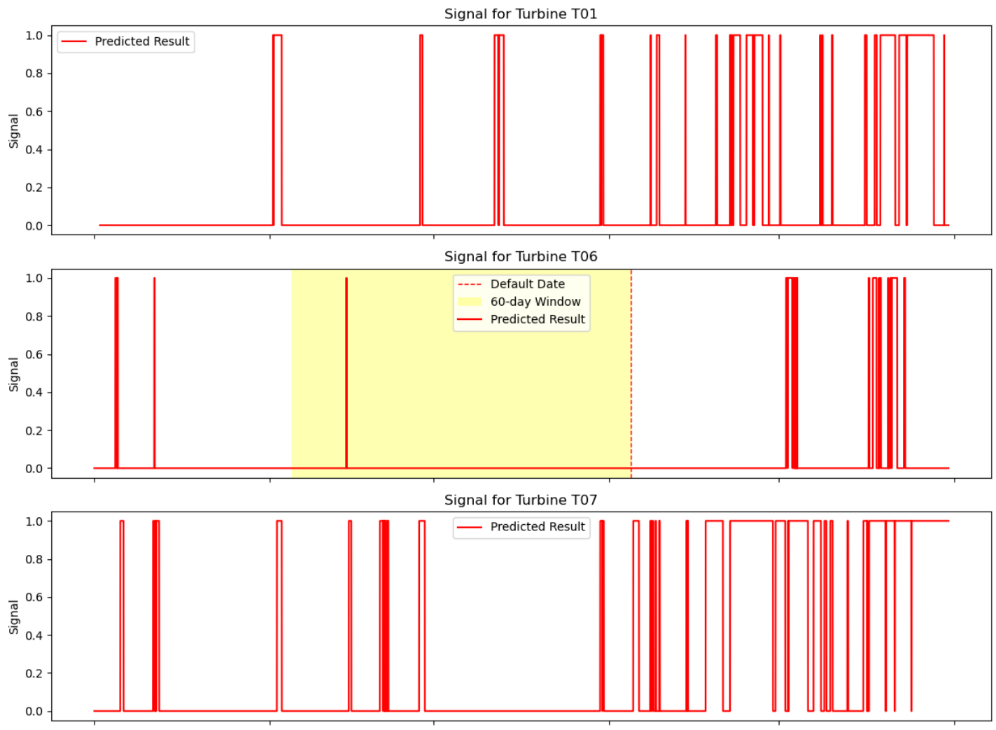

Figure 3-14 Signal Visualization of LSTM

We suspect that the root cause of these issues may be overfitting, potentially exacerbated by sample imbalance. Although we employed Synthetic Minority Over-sampling Technique (SMOTE) to balance the dataset, this method has its drawbacks. Specifically, if the synthetic samples generated by SMOTE do not accurately represent the true underlying distribution of the minority class, it can inadvertently lead to overfitting. This misrepresentation could be a significant factor in the suboptimal performance of our LSTM model.

4 Conclusion and Discussion

4.1 Summary of Results and Final Predictions

In the evaluation of predictive models, we observed a diversity in performance across several key metrics: accuracy, precision, recall, and F1 score. Taking the performance on validation set of hydraulic system as an example, the performances of the models are as summarized in Table 4-14.

The Isolation Forest achieves moderate accuracy (64.077%) but falls short in precision and recall. The SVM model, although less accurate overall (55.952%), is notably precise in its positive predictions (75.785%). The Vanilla Neural Network strikes a balance with reasonable accuracy (73.792%) and moderate precision and recall rates. In contrast, the LSTM model, while highly accurate (80.971%), has very low precision and recall, indicating a potential overemphasis on negative classifications.

These varied performances across models highlights the need for a tailored approach to model selection, where accuracy must be balanced with the ability to correctly identify and capture true positive cases for effective maintenance interventions.

Table 4-14 Evaluation of Performances on Validation Set of Hydraulic System

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
Isolation Forest	64.077	9.23	5.2	6.6
SVM	55.952	75.785	56.094	45.653
Vanilla Neural Network	73.792	44.792	32.242	37.495
LSTM	80.971	1.263	1.710	1.453

The final predictions of our project were generated using the Vanilla Neural Network, leveraging the entire dataset for training. This approach capitalizes on the comprehensive data to maximize learning and model accuracy. To mitigate the risk of overfitting, especially concerning the infrequent default events which could skew model performance, we implemented a strategic adjustment. Specifically, we gradually increased the dropout layer parameter across multiple iterations of the network. This technique introduces randomness into the training process, effectively helping the model generalize better by preventing it from relying too heavily on any single or small set of features.

We developed several variations of the Vanilla Neural Network, each with a different configuration of the dropout parameter. By comparing the predictions across these models, we adopted a consensus-based approach to determine the final predictions. This method enhances reliability, as the recurring predictions across multiple models are likely to be more robust and less prone to errors specific to any single model's idiosyncrasies.

Further validating our neural network approach, we compared its results with those obtained from other predictive models, such as SVM. This comparative analysis helps confirm the consistency and accuracy of our predictions, ensuring that the final periods identified as under suspicion of leading to turbine failure are recognized by multiple models, thereby increasing confidence in the predictive power and stability of our results.

Through this multifaceted modeling strategy, we not only increase the reliability of our predictive outcomes but also ensure a balanced approach that acknowledges and addresses the complexities inherent in predicting rare events in large datasets. The model's predictions on unseen test data from September 1, 2017, to December 31, 2017, resulted in total savings of 14,687.04 Euros, demonstrating a moderate capability to anticipate default events before they occurred.

4.2 Future Improvement Directions

For the challenge of wind turbine default signals, our primary obstacle is the extreme imbalance of our dataset, where default instances are rare. To tackle this, we've employed a range of techniques, ranging from classification to outlier detection, from linear regression, SVM, to more structured deep learning models. Our findings indicate that linear predictions are insufficient due to their simplicity, while more complex learning models pose a significant risk of overfitting. Among the approaches tested, the vanilla neural network emerged as the most effective, standing out for its capacity of capturing nonlinear correlations and simplicity within the deep learning spectrum.

Despite this success, we believe that further improvements can be achieved through the following directions:

- **Enhanced data handling strategies:** The next phase of the research could focus on selecting features that show a high correlation with the outcomes to reduce noise. Additionally, various sampling techniques could be deployed to manage the imbalance without introducing undue bias.
- **Neural Network Structure:** Ongoing efforts can be contributed to refining the predictive models. We have fine-tuned the hyperparameters and network structure, and there may still exist room for structural modification to meet with the specific properties of the datasets.
- **Boosting technique experimentation:** Techniques such as AdaBoost, Gradient Boosting Machines (GBM), and more sophisticated forms like XGBoost, LightGBM, and CatBoost may offer promising avenues. These models not only provide a framework for boosting weak learners but also include advanced features for handling data irregularities and complex interactions.

Appendix

GitHub Repo: EDP-Renewables-EarlyWarningSystem