Ensemble Methodology for Automated Machine Predictive Maintenance Classification

Corwin Fister
Electrical Engineering
University of Dayton
Dayton, United States
fisterc2@udayton.edu

Isaiah Hill
Electrical Engineering
University of Dayton
Dayton, United States
hilli3@udayton.edu

Timothy Marshall
Electrical Engineering
University of Dayton
Dayton, United States
marshallt6@udayton.edu

Aidan Picard
Electrical Engineering
University of Dayton
Dayton, United States
picarda1@udayton.edu

Theus Aspiras
Electrical and Computer Engineering
University of Dayton
Dayton, United States
aspirast1@udayton.edu

Abstract—Machine predictive maintenance is important for low-cost maintenance of manufacturing machines. Current methodologies use different classification models for predicting machine maintenance and failures. We propose an ensemble model that utilizes current state-of-the-art classification models for predictive maintenance: random forest, sequential neural network, XGBoost, decision tree, and support vector machine models.

I. INTRODUCTION

Machine failure is a very common problem in manufacturing that costs companies a lot of time and money due to machines becoming inoperable or requiring expensive components to get them back into production. Figure 1 illustrates that there is an optimal zone for maintaining machinery without incurring costly repairs during failure events. In light of this, many methodologies have been used for machine failure prediction. Every algorithm has its upsides and downsides, which have been individually researched and implemented.

We will be evaluating an ensemble model that encompasses the upsides of all the algorithms while minimizing the downsides, as shown in 2. This would prove beneficial to companies and corporations because the goal of machine failure prediction is to ensure that machines do not fail; false positives are less important than false negatives. This is because a false negative can allow a machine to fail without the maintenance teams even knowing that it is going to fail. Thus, the ensemble model that implements the other algorithms should have more false positives than false negatives. This is a key way to show that the machine failure prediction algorithm is doing its job effectively.

II. RELATED RESEARCH

Previous related research explores different types of algorithms that are best for machine maintenance. Preethi Bennet [1] found multiple classifiers such as gradient boosting, bagging, stacking, and random forest algorithms to be the
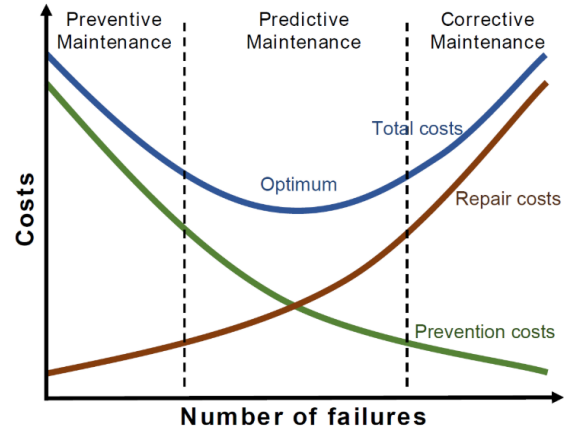

Fig. 1. Cost vs. Failure

best for predictive maintenance. Dharithri Sharma [2] found the random forest classifier algorithm to be the overall best algorithm for predictive maintenance on this dataset. Sara Amer [3]. found the random forest algorithm, the XGBoost algorithm, the decision tree algorithm, and the support vector machine algorithm to be the four best algorithms for predictive maintenance on this dataset. Both Dharithri Sharma [2] and Abdul Ismail [4] also found the random forest model to be the best overall model. Piyush [5] found the XGBoost algorithm to be the best overall algorithm. Overall, the best predictive maintenance algorithms, accuracy-wise, from these findings include the random forest algorithm, the XGBoost algorithm, the decision tree algorithm, the support vector machine algorithm, and the sequential neural network algorithm. It should be noted that the accuracies between each paper for these models differ due to differences in approach, specifically differences in the use of the dataset and the training of the
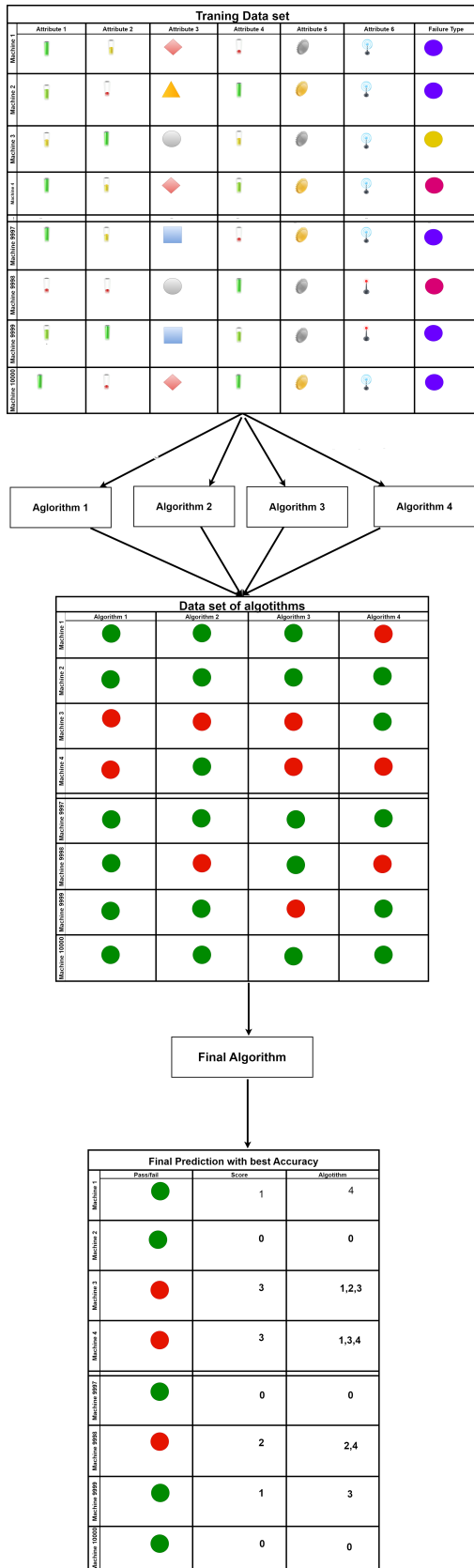
Fig. 2. Flowchart of Ensemble Algorithm

models.

Hau Tang [6] used different types of predictive maintenance algorithms. For instance, to extract features they used a convolutional neural network. This led to results similar to the previously mentioned algorithms, with slightly more accurate results; however, this could be due to over-fitting rather than true higher accuracy. Lala Rajaoaisoa [7] and Jovani Dalzochio [8] used some of the same models as Hau Tang [6]. Adem Avci [9] attempted to predict the remaining lifespan of different machines using a novel approach. This approach aimed to display how much time could be spent before maintenance must be done, thereby saving time and money on aircraft. Karolina Kudelina [10] used linear discriminant, linear SVM, KNN, wide neural network, and medium neural network models; however, they did not reach the same degree of accuracy as the previously mentioned models. This could be due to differences in the original dataset.

João Campos [11] used ensemble models to attempt to reach higher accuracy. However, their technique did not account for real-world use cases of these machine predictive maintenance algorithms and instead aimed at reaching the highest possible accuracy. Ram Mazumder [12] found that machine learning algorithms can be used for pipeline maintenance instead of physics models, which are expensive and energy-intensive. Bram Steurtewagen and Dirk Van den Poel [13] found that machine learning on sensor data can make predictive maintenance easier and more understandable. Ram K. Mazumder [12] found that XGBoost is the best algorithm for pipeline failure prediction.

In research done by Yildirim [14], they used the same dataset that we performed our analysis on. They referred to their work as anomaly detection. Despite this slight difference in description, the goal is ultimately the same: predicting machine maintenance and foreseeing anomalies in machines. In their research, Yildirim [14] examined two scenarios: first correlating two features and then correlating all the features. For the former, K-Nearest Neighbors was the best algorithm, with gradient boosting and the Extra Trees algorithm also performing well. For the latter, Random Forest was found to be the most accurate, with gradient boosting and AdaBoost also performing well.

Yuan [15], in their research, used the Random Forest, which has been a recurring theme in most of our research up to this point; however, they applied some changes. They used hierarchical clustering with the Dunn index and k-methods to make Random Forest more diverse. They performed this on a dataset classifying indoor temperature preferences.

In Bhekisipho Twala's paper: "Software Faults Prediction Using Multiple Classifiers" [16], they attempted to predict software faults as opposed to future machine maintenance, like we are doing in our work. Twala [16] used two of NASA's public datasets and performed a series of different algorithms, concluding that ensemble learning methods that include association rules or decision trees increase the prediction accuracy of basic classification algorithms.

We also found a paper written by K. Mishra and S. K.

Manjhi [17] who attempted to build a machine-learning model that would take into account service tickets, error codes related to a component of the device, inventory of the device, installation date, age of the device, number of transactions to date for a particular device, type of transactions performed, weather-related factors, and more to help ATMs function more efficiently. With this goal in mind, they laid out a proper data flow and architecture layout for such a machine-learning algorithm. These codes were then translated into the attributes of a data frame to which the Random Forest algorithm was applied. This paper contained some similarities to one written by Daniel Jung et al. [18], who hoped to find some way to implement predictive maintenance in the automotive industry. In his paper, Jung composed a fault classification system algorithm for time-series data.

In work by Irrerra [19], a support vector machine failure classifier was used to protect an Apache Tomcat web server. This predictor was able to function even through changes in the targeting system. The support vector machine, while not often the most accurate, was able to accomplish the task in this scenario.

Ilyasov [20] proposed a three-step method using the Gradient Booster algorithm. In the first step, sample data and a collection of features are created. In the second stage, the data is preprocessed—this includes filling in missing values, coding features by categories, and removing outliers. Finally, in the third step, the Gradient Booster Algorithm is applied to the dataset to anticipate the failure of the equipment, which in their case is deep-pumping equipment.

In a paper by Jing Li [21], a classification tree model was created that performs better than some state-of-the-art models in terms of accuracy, stability, and interpretability. They ran a test on different families of hard drives and, even with a smaller group of drives, found an algorithm that outperformed. Their classification tree model consisted of ten attributes and used Information Gain as the splitting function.

In research done by Natthakarn Limprasert [22], they walk through steps to apply a machine learning approach to solving the issue of predictive maintenance. These steps are as follows: gathering data, preparing the data, choosing the algorithm, training the model, evaluating the model, hyperparameter tuning, and achieving the best accuracy. After examining five different algorithms (random forest, decision tree, logistic regression, K-nearest neighbor, and support vector machine), they found that the most accurate one was the random forest algorithm, which matches several papers we have examined.

We also examined another paper written by Jalali [23] that attempted to predict the time remaining before the failure of plasma etching equipment. To do this, they first predicted the time-to-failure trend using linear regression. Then, they transformed the target variable to create a state prediction. Lastly, they showed the time to failure using classifications such as support vector machines, decision trees, random forest, multilayer perceptrons, gradient boosting classifier, SVM with stochastic gradient descent, and K-nearest neighbors.

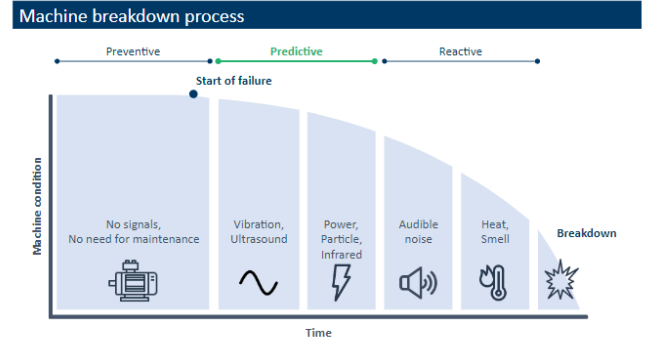Overall, these previous researchers provide two key high-

Fig. 3. Machine breakdown over time

lights for machine failure prediction. First, they show that the random forest model, the sequential neural network model, the decision tree model, the XGBoost model, and the SVM model are the best models for predictive maintenance. Second, they highlight a gap in the research. There is little research on what to do for the most optimal results in terms of saving money in industry. In manufacturing environments where machine failure would cost money and production time, one prefers the machine prediction algorithm to find more false positives than actual positives, so no false negatives occur. This is to ensure that the expensive machinery does not break and cost the company that owns the equipment thousands of dollars. Thus, one would prefer to perform more maintenance checks than necessary rather than fewer than necessary.

III. METHODOLOGY

The dataset being used is a simulated dataset [24]. In other words, it does not truly come from real-world machines. Instead, it is created synthetically for failure prediction analysis on CNC machines.

Figure 3 shows the machine breakdown process of machinery, which are based on various sensor inputs to provide types of machine breakdowns.

For pre-processing, we confirmed all columns and rows had data without duplicates or errors. We initially checked for outliers and correlations using pair plots and heatmaps. We then started to organize the data by failure type and explored scatterplots of categories that seemed to show some correlation. The columns included in the dataset are: 'UDI', 'Product ID', 'Type', 'Air temperature [K]', 'Process temperature [K]', 'Rotational speed [rpm]', 'Torque [Nm]', 'Tool wear [min]', and 'Machine failure' ('Target' in other data sets), and individual columns for each failure type that just indicate if the machine failed or not. The 'Machine Failure' column shows if a machine has failed, in similar datasets this would be the 'Target' Column. The types of failures include 'TWF' (Tool Wear Failure), 'HDF'(Heat Dissipation Failure), 'PWF' (Power Failure), 'OSF' (Overstrain Failure), and 'RNF'(Random Failure). In similar data sets the individual columns for the type of machine failure are all put into a single

column showing each failure type. UDI is the index number plus one. and product ID is the individual serial number of the machine. Type is the type of machine: 'H' (High), 'M' (Medium), and 'L' (Low). Air temperature measured in Kelvin, Process Temperature also measured in Kelvin, Rotational Speed in rotations per minute, Torque measured in Newton-meters, and Tool Wear in minutes. Machine failure or Target is simply the column that resembles machine failure as 1 and machine non-failure as 0. There are 10000 different rows of data representing 10000 individual machines.

From pre-processing, we concluded that the Product ID and UDI are just columns to keep track of the index of each machine with its individual identification code, and the machine type shows very little correlation to anything so we would like to not include these in our algorithms. The other columns are all important numerical columns that will be imported into any of the algorithms used.

There are multiple different machine failure columns. The 'TWF' column stands for tool wear failure. The 'HDF' column stands for heat dissipation failure. The 'PWF' column stands for power failure. The 'OSF' column stands for overstrain failure. The RNF column stands for random failure. Finally, the 'Machine Failure' column stands for machine failure. None of these machine failure columns are inputted into the algorithms. The descriptive numerical columns including air temperature, process temperature, rotational speed, torque, and tool wear are all inputted.

Random forest, neural network, XGB, decision tree, and SVM algorithms are implemented and used on the dataset. These are well-known methods previously used for predictive maintenance. All the methods are combined by adding them together. Failure is represented as 1 and non-failure is represented as 0. This method only works to save money and time if the assumption that false positives are allowed and do not lose money and time holds.

A. Classification Algorithms

Several different algorithms are used including random forest, sequential neural networks, XGB, and decision trees. Every one of these algorithms has pros and cons. By using multiple, the pros of each can be emphasized and the cons can be reduced.

Random forest is a method that selects random observations, builds decision trees based on those observations, and then does majority voting to decide the algorithm's output. The algorithm essentially acts as a majority voting system. It is like having a team of observers in the data set deciding what to do.

Sequential neural networks have layers of nodes that lead to the result. Each node applies weights multiplied by inputs and a bias to add the final result of the sum of the multiplication of weights and inputs. There are connections between every layer. For example, layer one's neurons are connected to every neuron in layer two. Then, the neural network applies a learning algorithm to modify its weights and biases to optimize

Model	Accuracy
Ensemble Model	0.989
Random Forest Model	0.991
Sequential Neural Network Model	0.983
Decision Tree Model	0.986
XGBoost Model	0.988
SVN Model	0.983

the neural network. The neural network implemented uses the Adam optimizer as the learning algorithm.

XGB is a machine learning algorithm that uses ensemble learning. It is based on a neural network. It fits a regression tree to the residuals. This regression tree is a special type of tree that is based on similarity scores. Similarity scores are defined as

$$\frac{(\sum Residuals)^2}{ResidualCount + \lambda}. \tag{1}$$

Output values on the tree are defined as

$$\frac{\sum Residuals}{ResidualCount + \lambda}. \tag{2}$$

Decision trees are hierarchical models that use a tree-like model of decisions and their consequences to classify data into different classes. Decision trees are useful for classification and categorization; therefore, it is good for classifying machine failure or machine non-failure.

IV. RESULTS AND DISCUSSION

Two types of accuracy are calculated in this paper. The first accuracy calculation excludes data points where the actual machine failure value is 0. This accuracy is found using the equation:

$$Accuracy = 1 - \frac{\sum \frac{\left| \sum \frac{Result}{5} - Actual \right|}{Actual}}{Count}. \tag{3}$$

where the five represents the five different models used, and the count is the number of data points, which is 2801. The accuracy using this equation is 0.986.

The second accuracy is calculated using:

$$Accuracy = 1 - \frac{\sum \frac{\left| \sum \frac{Result}{5} - Actual \right|}{|Actual| + 1}}{Count}. \tag{4}$$

This is more accurate because it accounts for values that are 0. For this equation, all values in the ensemble model greater than or equal to 3 are set to 5 to indicate that they need maintenance. The accuracies of the prediction models on their own are shown in Table I.

As shown in Table I, the random forest model is slightly more accurate than the ensemble model; however, the ensemble model allows for the maintenance team to see different machines that might break soon that are predicted by other models, thus making it better at the task at hand. This is because the ensemble model gives a value from 0 to 5

5371	0	0	0	0	0	0	0
2160	0	0	0	0	0	0	0
797	0	0	0	0	0	0	0
4302	0	1	0	1	1	0	3
526	0	0	0	0	0	0	0
3750	0	0	0	0	0	0	0
3626	0	0	0	0	0	0	0
9701	0	0	0	0	0	0	0
6763	0	0	0	0	0	0	0
9014	1	1	0	1	1	0	3
3872	0	0	0	0	0	0	0
2192	0	0	0	0	0	0	0
5969	0	0	0	0	0	0	0
1627	0	0	0	0	0	0	0
5522	0	0	0	0	0	0	0
1032	0	0	0	0	0	0	0
5599	0	0	0	0	0	0	0
2384	0	0	0	0	0	0	0
6849	0	0	0	0	0	0	0
1049	0	0	0	0	0	0	0
1290	0	0	0	0	0	0	0
2333	0	0	0	0	0	0	0
1329	0	0	0	0	0	0	0
4249	1	0	0	1	1	0	2

Fig. 4. Each algorithms prediction with a total sum of predicted failures

depending on how many models showed that it is accurate. It can also be looked at from a view where one sees what model gives which results, thus allowing for more complex machine prediction.

The sequential neural network model and the support vector machine model are the least accurate of the five models used in the ensemble model. This means that they predict the least accurately among the five models used. The neural network model gives many false positives, and the support vector machine model gives many false negatives. This would make the neural network better for predictive maintenance because false positives are not as serious as false negatives in machine prediction. This is because doing more maintenance than necessary so that expensive machinery does not break is not nearly as bad as not doing maintenance that will break expensive machinery.

The most accurate model is the random forest model; however, it often has false negatives which are not good for predictive maintenance because of the seriousness of false negatives. This makes the ensemble model the best to use. This is because, in the ensemble model, one gets the best of all worlds.

The decision tree model and the XGB model are in between the best and the worst models. They have both many false positives and many false negatives. This means that they are relatively good at predicting machine failure.

A. Discussion

The ensemble model displays several key strengths that the other models do not. Firstly, it catches more machine failures than the other models by utilizing their predictions and making its own. Secondly, it can indicate that certain machines might fail, allowing a maintenance specialist to check the machine before it fails completely. When a machine fails, it can take time for parts to be ordered, repairs will take longer as diagnostics are needed, and the machine will be out of production until the repair is made.

Being able to predict when a machine will fail is beneficial as the downtime for the machine could be planned and reduced to increase efficiency. When a machine fails unpredictably, it can cause many problems with safety and efficiency. If a machine is predicted to fail, it can be shut down before it causes any injuries, and then a technician can inspect and service it. When an unexpected failure occurs, the technician able to service it may be maintaining or servicing other equipment. After an inspection or diagnostic is done, parts will need to be ordered, which may not be immediately available.

The ensemble model's ability to catch certain failures that other models do not demonstrates its usefulness. The decision tree and XGBoost models prove insufficient for the problem at hand because of their false positives and false negatives. The random forest model offers many false negatives, which is problematic for predictive maintenance because false negatives do not predict whether a machine will fail. In contrast, false positives can lead to more maintenance but will save time and money in the long run. The sequential neural network has high accuracy; however, it misses key machine failures that, if caught, would save money. Thus, the ensemble model proves to be the most effective for predictive maintenance.

V. Conclusion

Using the ensemble model, which encompasses the upsides of all the algorithms while minimizing the downsides, proves beneficial to companies and corporations. It reduces repair costs and increases efficiency because the goal of machine failure prediction is to ensure that machines do not fail. In this context, false positives are less important than false negatives. This is because a false negative can allow a machine to fail without the maintenance teams even knowing it is going to fail. Thus, the ensemble model, which implements multiple algorithms, should have more false positives than false negatives. This demonstrates that the machine failure prediction algorithm is performing its job effectively.

While some failures will still remain unpredictable, we can minimize their occurrence. By combining the best techniques for predicting failure into one algorithm, the total cost of maintenance should be reduced, and efficiency increased.

References

[1] D. Yung, "Data-driven open-set fault classification of residual data using bayesian filtering," in *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY*, vol. 28, no. 5, 2020.

[2] D. B. Sharma, Sripradha, Nikita, A. Kodipalli, T. Rao, and R. B R, "Machine predictive maintenance classification using machine learning," in *2023 International Conference on Computational Intelligence for Information, Security and Communication Applications (CIISCA)*, 2023, pp. 308–313.

[3] S. Amer, H. k. Mohamed, and M. Badr Monir Mansour, "Predictive maintenance by machine learning methods," in *2023 Eleventh International Conference on Intelligent Computing and Information Systems (ICICIS)*, 2023, pp. 58–66.

[4] A. S. Ismail and F. A. Abdulazeez, "Leveraging machine learning algorithms for predictive maintenance in internet of things (iot) systems," in *2023 12th International Conference on System Modeling Advancement in Research Trends (SMART)*, 2023, pp. 271–274.

[5] Piyush, R. Kumar, N. Goomer, H. S. Rana, S. Chhabra, and A. Singh, "Predictive maintenance for industrial equipments using ml dl," in *2023 International Conference on Advanced Computing Communication Technologies (ICACCTech)*, 2023, pp. 391–396.

[6] H. Tang, X. Cui, J. Zhu, L. Shen, J. Cai, and B. Zhu, "A bayesian-optimized hybrid neural network based on cnn and bilstm for predictive maintenance of diesel generator," in *2023 Prognostics and Health Management Conference (PHM)*, 2023, pp. 17–22.

[7] L. Rajaoarisoa, R. Randrianandraina, and M. Sayed-Mouchaweh, "Predictive maintenance model-based on multi-stage neural network systems for wind turbines," in *2024 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA)*, 2024, pp. 1–7.

[8] J. Dalzochio, R. Kunst, J. Barbosa, E. Pignaton De Freitas, and A. Binotto, "Milpdm: A predictive maintenance architecture for the military domain," in *2023 International Conference on Machine Learning and Applications (ICMLA)*, 2023, pp. 2026–2032.

[9] A. Avci and N. Acir, "Remaining useful life estimation with parallel convolutional neural networks on predictive maintenance applications," in *2020 28th Signal Processing and Communications Applications Conference (SIU)*, 2020, pp. 1–4.

[10] K. Kudelina, H. A. Raja, S. Autsou, M. U. Naseer, T. Vaimann, A. Kallaste, R. Pomamacki, and V. K. Hyunh, "Preliminary analysis of mechanical bearing faults for predictive maintenance of electrical machines," in *2023 IEEE 14th International Symposium on Diagnostics for Electrical Machines, Power Electronics and Drives (SDEMPED)*, 2023, pp. 430–435.

[11] J. R. Campos, E. Costa, and M. Vieira, "Improving failure prediction by ensembling the decisions of machine learning models: A case study," *IEEE Access*, vol. 7, pp. 177 661–177 674, 2019.

[12] R. K. Mazumder, A. M. Salman, and Y. Li, "Failure risk analysis of pipelines using data-driven machine learning algorithms," *Structural Safety*, vol. 89, p. 102047, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167473020301259

[13] B. Steurtewagen and D. Van den Poel, "Adding interpretability to predictive maintenance by machine learning on sensor data," *Computers & Chemical Engineering*, vol. 152, p. 107381, 2021.

[14] B. Steurtewagen and D. Van den Poel, "Adding interpretability to predictive maintenance by machine learning on sensor data," *Computers Chemical Engineering*, vol. 152, p. 107381, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0098135421001599

[15] H. Yuan, X.-S. Fan, Y. Jin, J.-X. He, Y. Gui, L.-Y. Song, Y. Song, Q. Sun, and W. Chen, "Development of heart failure risk prediction models based on a multi-marker approach using random forest algorithms," *Chinese medical journal*, vol. 132, no. 07, pp. 819–826, 2019.

[16] B. Twala, "Software faults prediction using multiple classifiers," in *2011 3rd International Conference on Computer Research and Development*, vol. 4. IEEE, 2011, pp. 504–510.

[17] B. Yildirim, H. F. Alsan, and T. Arsan, "Machine failure prediction: : A comparative anomaly detection," in *2023 Innovations in Intelligent Systems and Applications Conference (ASYU)*, 2023, pp. 1–6.

[18] J. Li, X. Ji, Y. Jia, B. Zhu, G. Wang, Z. Li, and X. Liu, "Hard drive failure prediction using classification and regression trees," in *2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, 2014, pp. 383–394.

[19] D. Yuan, J. Huang, X. Yang, and J. Cui, "Improved random forest classification approach based on hybrid clustering selection," in *2020 Chinese Automation Congress (CAC)*, 2020, pp. 1559–1563.

[20] B. Twala, "Software faults prediction using multiple classifiers," in *2011 3rd International Conference on Computer Research and Development*, vol. 4, 2011, pp. 504–510.

[21] K. Mishra and S. K. Manjhi, "Failure prediction model for predictive maintenance," in *2018 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*, 2018, pp. 72–75.

[22] I. Irrera, M. Vieira, and J. Duraes, "Adaptive failure prediction for computer systems: A framework and a case study," in *2015 IEEE 16th International Symposium on High Assurance Systems Engineering*, 2015, pp. 142–149.

[23] B. Ilyasov, E. Makarova, V. Martynov, E. Zakieva, E. Gabdullina, and M. Yusupov, "Application of gradient boosting algorithm for predicting equipment failures," in *2022 VI International Conference on Information Technologies in Engineering Education (Inforino)*, 2022, pp. 1–5.

[24] "Machine Failure Predictions, howpublished = https://www.kaggle.com/datasets/shashanknecrothapa/machine-failure-predictions/data, note = Accessed: 2024-01-06."