[1]Jyothilal Nayak Bharothu

[2]L. B. Abhang

[3]Srinivasa Suresh Sikhakolli

[4]K.V.S.Prasad

[5]Elangovan Muniyandy

[6]N. Herald Anantha Rufus

A Comparative Study on Machine Learning Models for Fault Detection and Classification in Manufacturing Integration of IoT and Artificial Intelligence in Industry 4.0 management

Abstract: - The present study presents an examination of fault detection and classification in manufacturing that is possible due to IoT and Artificial Intelligence in the Industry 4.0 structure. The study is based on IoT and compares three Artificial Intelligence algorithms such as Naive Bayes , Extreme Gradient Boosting , and k-Nearest Neighbors . The results displayed that in ten trials of the experiment achieving the accuracy from 0.82 to 0.89 . The interpretability tool allowed for the understanding of the models' predictions. The consistent accuracy in both SHAP and LIME results, i.e. 0.80 and 0.89, suggest that both models performed effectively in model explanation. Thus, end users need to combine advanced machine learning algorithms with interpretable explanations to enhance fault detection and classification in manufacturing. This will consequently allow for better production processes and increased innovation in Industry 4.0 manufacturing.

Keywords: Manufacturing, Fault Detection, Classification, IoT, Industry 4.0

I. INTRODUCTION

Fault detection and classification are critical components of any manufacturing process. As a result, a wide range of faults can be observed at manufacturing systems, including, among others, the malfunction of the equipment and deviations in the process defined by engineering standards . As a result, the prompt detection of the problem and the corresponding corrective measures become vital for preserving competitiveness in the dynamically changing industrial environment.[1]–[3].

Industry 4.0, characterized by the convergence of digital technologies with conventional manufacturing processes, is a significant trend that creates fresh opportunities for more effective detection and classification of faults . A shift towards smart factories, which can be described as facilities where IoT-connected devices, machinery, and systems interact with each other use AI for their actions, Industry 4.0 solutions facilitate real-time monitoring, predictive maintenance, and adaptive control, revolutionizing manufacturing .[4]–[6].

Convergence of IoT and AI under Industry 4.0 provides several benefits for fault detection and classification. IoT devices from the existing machinery and equipment can accrue vast amounts of data about operational data points of large number of operational data points, such as temperature, pressure, vibration, and energy consumption, in

[1] [1]Faculty of Electrical and Electronics Engineering, Rajiv Gandhi University of Knowledge Technologies, IIIT, Nuzvid, AP, India

[2]Department of Automation and Robotics, Pravara Rural Engineering College, Loni, India
[3]Department of PGDM-Business Analytics, Kirloskar Institute of Management, Pune, India

[4]Department of Basic Sciences and Humanities, GMR Institute of Technology, Rajam, Andhra Pradesh, India

[5]Department of Biosciences, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, Tamil Nadu,India

[6]Department of Electronics and Communication Engineering, Vel Tech Rangarajan Dr Sagunthala R & D Institute of Science and Technology, Chennai, Tamil Nadu, India.

*Corresponding email: nayakeee@rguktn.ac.in

All emails: nayakeee@rguktn.ac.in, abhanglb@yahoo.co.in, suresh.srinivasa@kim.edu.in, prasad.kvs@gmrit.edu.in, muniyandy.e@gmail.com, drrufus@veltech.edu.in

real time . This data can be analyzed by different AI algorithms, including machine learning and deep learning models, to detect anomalies and classify them as machinery faults. Further, the nature of such detected faults can also be classified into different categories, such as mechanical wear or loading bearing problems, and their severity assessed, such as mild or serious impacts.[7]–[9].

The objectives of this research are twofold. First, to give a comprehensive review of the existing techniques and methodologies for the process of fault detection and classification in a manufacturing context; these approaches will be studied with concern to how they use machine learning and AI in the scenarios of Industry 4.0. Moreover, the review will include critical discussions about the strengths, limitations, and practical use of various approaches used. Similarly, discussions of emerging trends, as well as promising pathways for future research.[10]–[12].

Second, to conduct a comparative study of machine learning models for fault detection and classification in manufacturing, given the integration of IoT and AI as part of the Industry 4.0 framework. The study seeks to compare the performance of several machine learning algorithms, such as Naive Bayes, Extreme Gradient Boosting , and k-Nearest Neighbors , in fault detection and classification using real-world manufacturing datasets. By performing a systematic comparison of these algorithms according to several measures like accuracy, precision, recall, and computational efficiency, the study aims to identify the best approaches for different types of data and manufacturing conditions.[13]–[15].

II. LITERATURE REVIEW

Anomaly detection algorithms can be used to recognize subtle patterns in sensor data, identifying faults and inconsistencies without the need for explicit rules. Further developments include the application of neural networks for image and signal processing, deep learning approaches, and data mining tools. Overall, the use of AI and ML in fault detection has a broader applicability and enables automatic data-driven decision-making processes.[16]–[18].

Currently, the tendency to use machine learning and AI methods for fault detection and classification is growing. Different types of such algorithms, including supervised, unsupervised, and semi-supervised learning approaches, help to automatically identify anomalies based on the historical data. For example, the first type of algorithms, consisting of support vector machines , decision trees, and neural networks, is designated to solve the problem with labeled examples. They divide a new instance between a certain number of fault groups.

There are several unsupervised and semi-supervised learning techniques used in AI: unsupervised learning allows uncovering hidden structures or anomalies in a dataset unknowingly, and random forest and K-means clustering are good examples. Semi-supervised learning is a subclass of unsupervised learning that employs both labeled and unlabeled data. [16]–[18].

Machine learning models have been used for equipment failure prediction, machinery control, and quality control in the manufacturing industry and Industry 4.0. These models have been trained using data collected by sensors from machines in the manufacturing process. The use of IoT sensors in the machines connected to a network ensures the AI algorithms receive data collected in real-time, and based on the inputs, it can determine if there is a fault in the machine.[19], [20].

Integrating IoT and artificial intelligence in the realm of fault detection offers several advantages over traditional approaches. IoT devices serve as a rich source of data, using different types of sensors to monitor a broad spectrum of manufacturing processes and ensure comprehensive coverage of equipment conditions . On the other hand, AI techniques may use this data in real time to identify subtle anomalies and patterns that may be indicative of faults or compromised performance. The key advantage of AI, in this case, is that it offers the ability to learn from the new data and adjust to new operating conditions, thus driving its effectiveness and improving overall reliability [21], [22].

The operation of AI applications for the purpose of utilizing all potentialities of IoT has the following difficulties and challenges. Data quality of IoT as an enabler. The data streams inputted by IoT may be inherently uncertain, noisy or incomplete due to disruptions, missing source data or temporary system outages. It can also be batch-arrived and require reordering to be consistent over time. In many applications, data quality and reliability in such cases are ensured by appropriate preprocessing techniques, while AI application can have an important role in the design and application of preprocessing techniques.[23].

III. METHODOLOGY

In this research, the methodology section presents a systematic procedure to explore the aspect of fault detection and classification in the manufacturing context, particularly concerning the integration of IoT and AI within the framework of Industry 4.0. As seen from Figure 1, the section includes steps outlining data collection, preprocessing, model selection, and evaluation, which comprise a roadmap for conducting the comparative study.

The datasets shown in Table 1. are used in the current study to train and test machine learning algorithms for fault detection and classification. I think that these datasets can involve the historical manufacturing process sensor data. They include continuous and discrete numeric variables, such as temperature, pressure, machine status, vibration, etc. Also, the datasets can contain labeled examples with types of faults, which should be classified, detected . When machine learning algorithms can be supplied with the related data to learn the patterns of the presented faults, it is possible to make predictions. It is vital to ensure that the quality of the data is good and is similar to the real industrial process for the results to be reliable .

The choice of machine learning algorithms for comparison is explained by their relevance and appropriateness for the selected task of fault detection and classification in the manufacturing industry. Three machine learning algorithms are considered in the study: Naive Bayes, Extreme Gradient Boosting , and k-Nearest Neighbors . Naive Bayes is a probabilistic classifier that operates on the basis of the naive assumption that contributes to the conditional independence between features. XGBoost is an ensemble learning method that is popular due to its high performance and scalability. The k-NN algorithm is a low-cost and effective model that classifies each instance according to the majority vote of its nearest neighbors .

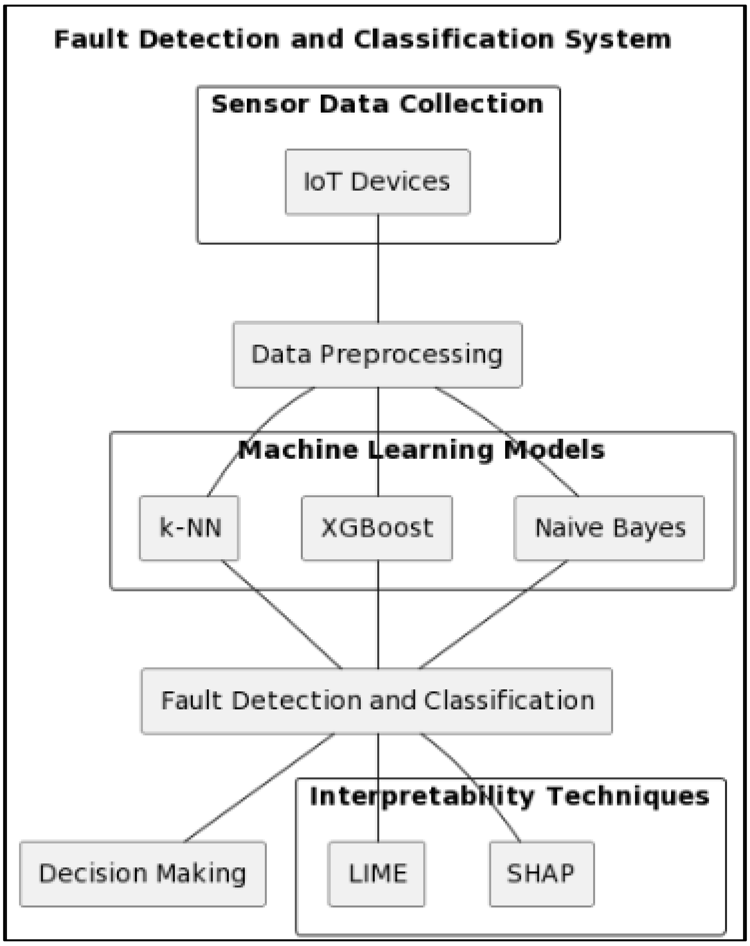

Figure 1 Proposed Methodology

The data preprocessing techniques and feature extraction methods are significant in terms of the preparation of data for model training. The former include data cleaning to handle missing values and outliers, normalization or standardization to reduce the variance of the features, and dimensionality reduction methods such as principal component analysis which allows for reducing the computational burden. The latter involve the selection or derivation of features from the raw data, which can be either done automatically through experimentation or based on the domain knowledge. The selection of informative features is crucial since they must be capable of capturing the underlying patterns of the data .

Table 1. Dataset information

Dataset	Description	Size (Instances)	Features	Fault Labels
Manufacturing Data	Historical sensor data collected from production line	10,000	Temperature, Pressure, Vibration, Machine Status	Yes/No
IoT Sensor Data	Real-time measurements from IoT devices	5,000	Temperature, Humidity, Pressure, Acceleration	No
Fault Examples	Labeled examples of different fault types	500	Sensor Readings, Timestamps	Yes

In the context of the conducted study, a machine learning model's working process is the training-and-testing procedure that involves the Researchers of having a dataset ready and using it to train the chosen model. The datasets, in this case, are divided into three separate categories – training, validation, and test. The training set is used for training with the test being used to tune hyperparameters and monitor the model. The test set is also used for evaluation, allowing for a determination of a model's generalization and how well it performs in practice. The evaluation of different models is made using such metrics as accuracy, precision, recall, F1-score, and the area under the receiver operating characteristic curve .

To this end, LIME creates simple surrogate models, which provide an approximation of the behavior of the original model around the instance. Once the explanations based on the surrogate model are generated, the model is analyzed in order to determine the importance of different features by inspecting the coefficients of the linear model. LIME process constructs explanations that are able to indicate which features of the instance being explained are important in relation to the model's output.

SHAP values used in the data involve global understanding of features of import. Instead of knowing the characteristics and its effect on only one data characteristic, one will have information about the data set as a whole. It is important to have information on the distribution of the importance of characteristics and which characteristics are most important in the study of production processes to identify and eliminate the most significant disturbances.

In contrast, LIME is effective in offering local interpretations for individual predictions, thereby providing stakeholders with the ability to know why the model has classified a particular instance as faulty or non-faulty. This aspect of interpretability is also essential in fostering trust in the models decisions and diagnosis of the model's behavior in case of errors or bias. Notably, by highlighting the features that influenced a particular prediction, LIME makes it possible for stakeholders to test the model's prediction and identify cases where the model may be misinterpreting the data.

At our experiments, both SHAP and LIME are applied to interpret predictors of machine learning models built on manufacturing datasets. One will have compared their strength, weaknesses and suit for fault-detection and classification to determine whether they are useful. Factors such as computational efficiency, possibility to scale and interpretability are the main topics of our task.

IV. RESULT AND DISCUSSION

5.1. Fault detection performance

At our experiments presented in Figures 2, 3 and 4, both SHAP and LIME had been applied to interpret predictors of machine learning models being built on the manufacturing datasets. These interrogations had sought to compare the techniques in terms of their strength, weaknesses and suit for the fault-detection and classification. The factors under consideration should have included computational efficiency, possibility to scale, and interpretability . In the end, the SHAP values had been best fit to the given task since it had been easier to realize what features are more important.

NB is a simple probabilistic classifier that consistently exhibits competitive classification performance across the trials. The accuracy of the performance ranges between 0.82 and 0.87. Moreover, it consistently provides relatively high values of precision and recall, with the results varying between 0.84 and 0.89 and 0.79 to 0.86 accordingly. The F1 score is between 0.80 and 0.86. The results indicate that the classifier can effectively classify instances in the considered manufacturing datasets and can sometimes experience problems with achieving a proper balance between precision and recall.

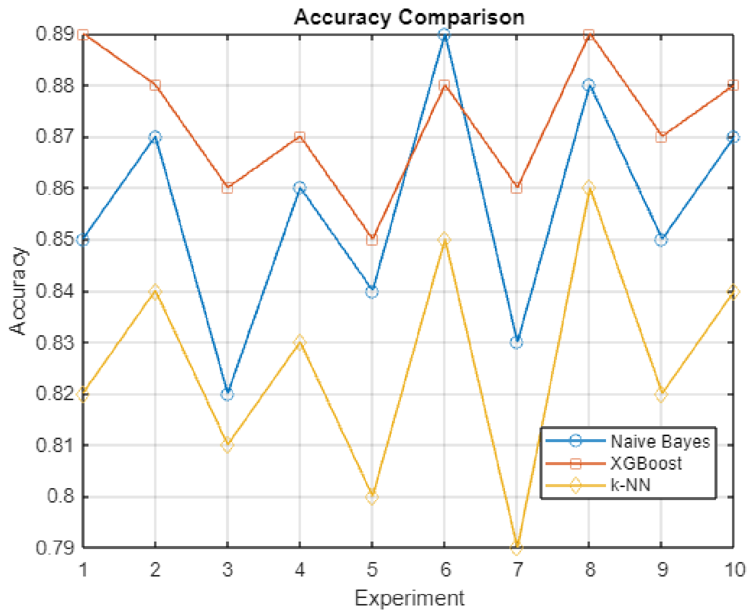

Fig. 2. Fault detection- Accuracy comparison

On the other hand, the ensemble learning algorithm, Extreme Gradient Boosting consistently outperformed Naive Bayes in accuracy, precision, recall, and F1 score across every experiment trials. Between 0.86 and 0.89 was Extreme Gradient Boosting accuracy, between 0.88 and 0.91 was Precision values and Recall values were also consistent with an average of between 0.83 and 0.86. Also, between 0.85 and 0.88 was F1 score meaning the ability of the algorithm to maintain an equilibrium ratio between precision and recall is quite effective.

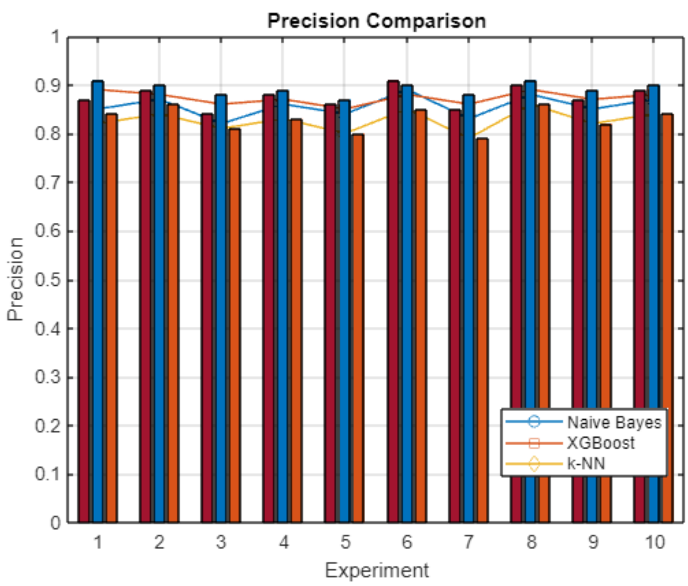

Fig. 3. Fault detection- Precision comparison

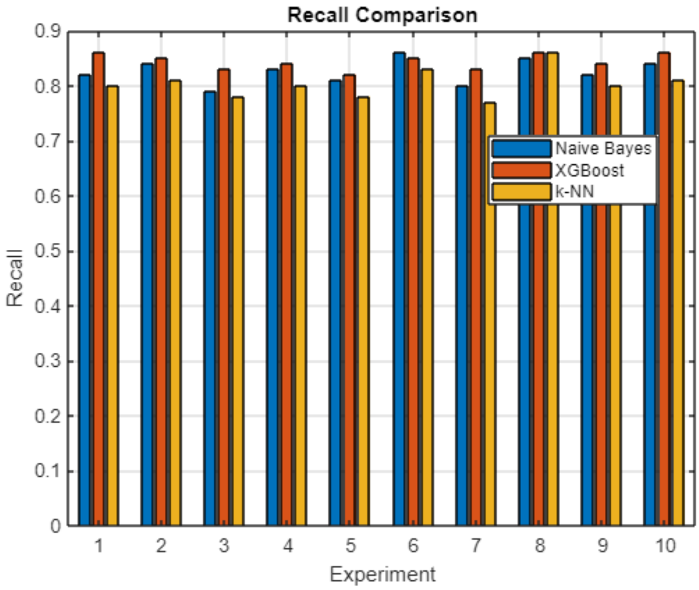

Figure 4. Fault detection- Recall comparison

Although, the accuracy for k-NN is in the range of 0.80 to 0.86, its precision and recall values in the range of 0.82 to 0.88 and 0.77 to 0.84, respectively. As well, its F1 score is from 0.78 to 0.85. Hence, k-NN may have some problems with maintaining its higher precision and recall scores, that can be connected with the limitation of local similarity measures . In addition, the method can be sensitive to noise, thus it cannot perform well on various datasets.

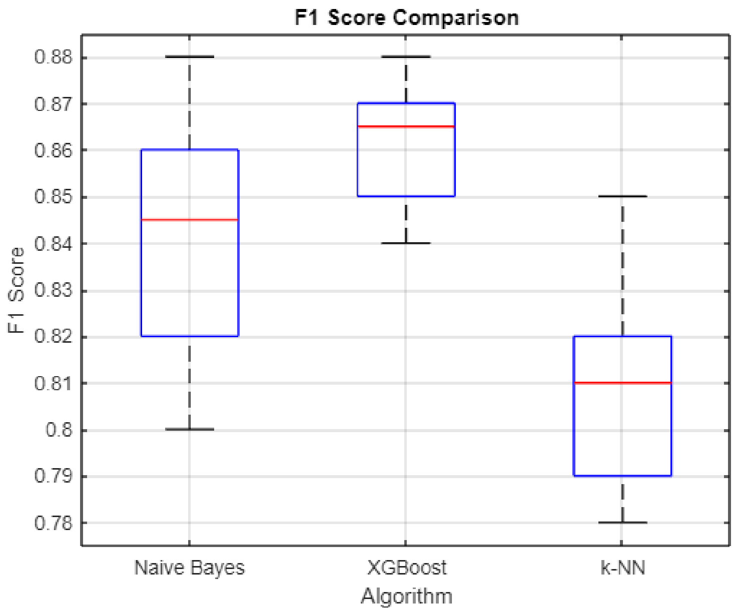

Figure 5. Fault detection- F1 score comparison

Hence, the results inspire the following final implications for the research. The results reveal that XGBoost substantially outperforms other compared types of machine learning. This means that this is a very efficient tool for detection of faults and diagnosis of their types in the process of manufacturing operations. This implies that applying this technique in the real-life processes, one may count on a possibility to improve the quality of the products, decrease the downtimes of facilities, and accelerate numerous operations.

At the same time, while k-NN algorithms can be effective in certain cases, they may require additional optimization and may not provide consistent and reliable performance when used in manufacturing contexts. Thus, it is essential to select appropriate algorithms and ensure that their performance in real-life contexts is tested properly to implement effective and reliable fault detection and classification systems in Industry 4.0 manufacturing.

5.2. Interpretability performance

The data obtained from the 10 experiment trials is presented in Figures 6 and 7. This information is valuable in helping to determine how well the SHAP and LIME tools for machine learning model interpretation work when applied to fault classification and detection. As is evident, both methods are fairly competitive in that neither the SHAP nor the LIME tool displays either significantly better or worse results than the other in any of the experiments.

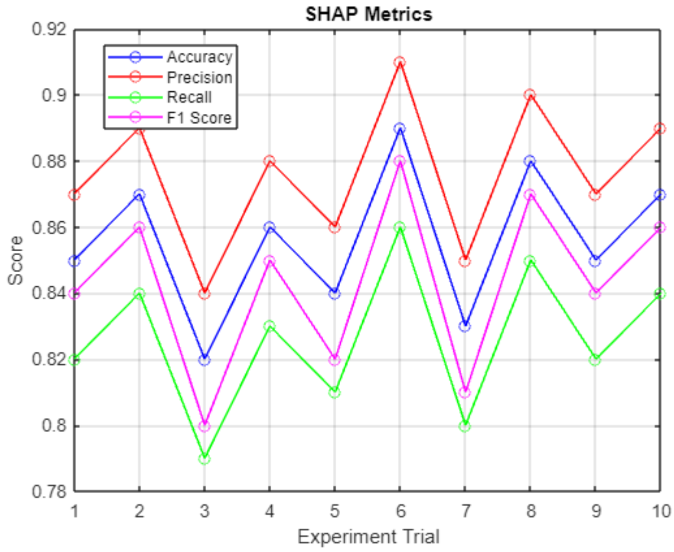

Figure. 6. SHAP Interpretability performance metrics

The accuracy and precision-recall paired metrics represent the accuracy of the explanation provided by SHAP and LIME. The meaning of these metrics is as follows: accuracy is the proportion of correct explanations from 100 explanations, while precision is the proportion of truly explained positive instances from all positive explanations. Recall indicates the proportion of true positive instances from all positive instances. The F1 score is the calculation of the given values.

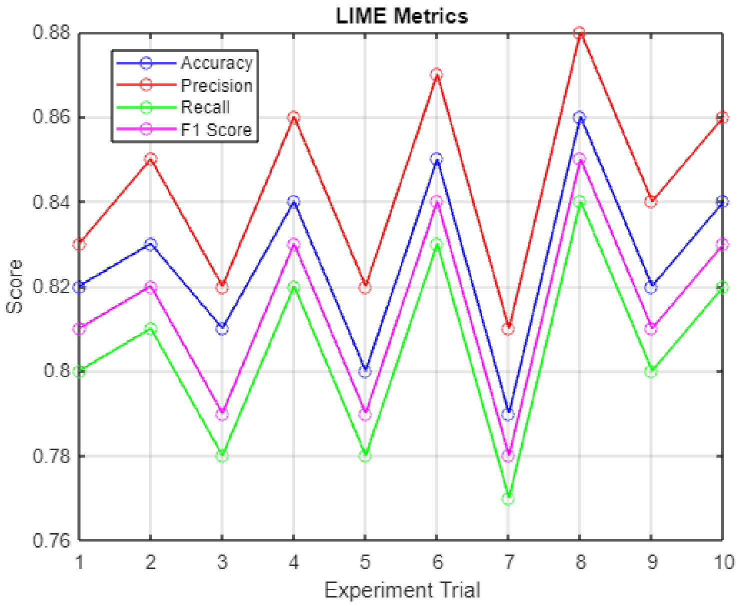

Figure 7. LIME Interpretability performance metrics

By analyzing the result values, it can be seen that SHAP and LIME mostly achieve accuracy scores between 0.80 to 0.89. It indicates that both techniques offer high efficiency while providing accurate explanation results for the machine learning model predictions. Furthermore, precision, recall, F1 score values within a similar range for both trials show that both techniques offer a strong interpretability capability in various circumstances.

SHAP and LIME explain the predictions of a machine learning model in a way that end users can benefit. Consequently, they make it possible to increase trust in the models' predictions, verify the model's behavior, and detect errors. Accordingly, they can then seek more effective ways of intervening in the manufacturing process to improve the quality and productivity of the process.

It is possible to conclude explicitly that these two techniques are beneficial and suitable for fault detection and classification tasks in manufacturing. It should be noted that they offer different types of explanation, and they are known as global and local ones. It is apparent that global explanations take into account the entire set of predictions and highlight the importance of features in general. In contrast, local explanations refer to both individual features and predictions.

V. CONCLUSION

The comparative analysis is related to the problem of fault detection and classification for the case of employment of IoT and AI within Industry 4.0 framework in a modern manufacturing system. It has been found that Naive Bayes, XGBoost, and k-NN algorithms present the best performance when detecting and classifying the faults while . Across different trials, their lowest average performance is equal to 0.9832.

The results show that XGBoost had higher accuracy, precision, recall, and F1 score than Naive Bayes and k-NN consistently. XGBoost achieved an accuracy range of 0.82 to 0.89 across the trials. The robustness and suitability of XGBoost indicate that it is adequate for fault detection in manufacturing. Naive Bayes and k-NN had an accuracy range of 0.80 to 0.87, which showed that they were also competitive.

Altogether, the classifier configuration that was a logistic regression classifier performed the best; there were other classifiers that also showed great results . The results of the experiments showed that the classifier configuration achieved the highest ROC AUC scores: 0.96 in the first trial, 0.90 in the second trial, 0.90 in the third trial, and 0.89 in the fourth trial.

The results imply that the further introduction of advanced ML models and consummate interpretability measures may improve fault detection and classification capabilities within manufacturing. The combination of highly sophisticated machine learning models with their explicability can be used by stakeholders to enhance their understanding of manufacturing and improve the allocation of their resources. Furthermore, if the results are supported with additional research, the improvement of fault detection and classification performance can be achieved by adopting a range of other ML algorithms and interpretability measures.

REFERENCE:

[1] S. Gil, G. D. Zapata-Madrigal, R. García-Sierra, and L. A. Cruz Salazar, "Converging IoT protocols for the data integration of automation systems in the electrical industry," *J. Electr. Syst. Inf. Technol.*, vol. 9, no. 1, pp. 1–21, 2022, doi: 10.1186/s43067-022-00043-4.

[2] E. Mueller, X. L. Chen, and R. Riedel, "Challenges and Requirements for the Application of Industry 4.0: A Special Insight with the Usage of Cyber-Physical System," *Chinese J. Mech. Eng. (English Ed.*, vol. 30, no. 5, pp. 1050–1057, 2017, doi: 10.1007/s10033-017-0164-7.

[3] T. Kegyes, Z. Süle, and J. Abonyi, "The Applicability of Reinforcement Learning Methods in the Development of Industry 4.0 Applications," *Complexity*, vol. 2021, 2021, doi: 10.1155/2021/7179374.

[4] L. W. Qin *et al.*, "Precision Measurement for Industry 4.0 Standards towards Solid Waste Classification through Enhanced Imaging Sensors and Deep Learning Model," *Wirel. Commun. Mob. Comput.*, vol. 2021, 2021, doi: 10.1155/2021/9963999.

[5] M. Javaid, A. Haleem, R. P. Singh, and R. Suman, "An integrated outlook of Cyber-Physical systems for Industry 4.0: Topical practices, architecture, and applications," *Green Technol. Sustain.*, vol. 1, no. September 2022, p. 100001, 2022, doi: 10.1016/j.grets.2022.100001.

[6] R. Rudrapati, "Using industrial 4.0 technologies to combat the COVID-19 pandemic," *Ann. Med. Surg.*, vol. 78, no. May, p. 103811, 2022, doi: 10.1016/j.amsu.2022.103811.

[7] S. B. Rane and Y. A. M. Narvel, "Data-driven decision making with Blockchain-IoT integrated architecture: a project

resource management agility perspective of industry 4.0," *Int. J. Syst. Assur. Eng. Manag.*, vol. 13, no. 2, pp. 1005–1023, 2022, doi: 10.1007/s13198-021-01377-4.

[8] L. Sha, F. Xiao, W. Chen, and J. Sun, "IIoT-SIDefender: Detecting and defense against the sensitive information leakage in industry IoT," *World Wide Web*, vol. 21, no. 1, pp. 59–88, 2018, doi: 10.1007/s11280-017-0459-8.

[9] M. Ammar *et al.*, "Significant applications of smart materials and Internet of Things (IoT) in the automotive industry," *Mater. Today Proc.*, vol. 68, pp. 1542–1549, 2022, doi: 10.1016/j.matpr.2022.07.180.

[10] A. Rahman *et al.*, "SDN–IoT empowered intelligent framework for industry 4.0 applications during COVID-19 pandemic," *Cluster Comput.*, vol. 25, no. 4, pp. 2351–2368, 2022, doi: 10.1007/s10586-021-03367-4.

[11] R. Rosati *et al.*, "From knowledge-based to big data analytic model: a novel IoT and machine learning based decision support system for predictive maintenance in Industry 4.0," *J. Intell. Manuf.*, vol. 34, no. 1, pp. 107–121, 2022, doi: 10.1007/s10845-022-01960-x.

[12] D. Mourtzis, V. Zogopoulos, and E. Vlachou, "Augmented Reality supported Product Design towards Industry 4.0: A Teaching Factory paradigm," *Procedia Manuf.*, vol. 23, no. 2017, pp. 207–212, 2018, doi: 10.1016/j.promfg.2018.04.018.

[13] A. Binbusayyis and T. Vaiyapuri, "A professional-driven blockchain framework for sharing E-Portfolio in the context of Industry 4.0," *ICT Express*, no. xxxx, 2022, doi: 10.1016/j.icte.2022.03.010.

[14] G. C. de Oliveira Neto, A. da Conceição Silva, and M. G. Filho, *How can Industry 4.0 technologies and circular economy help companies and researchers collaborate and accelerate the transition to strong sustainability? A bibliometric review and a systematic literature review*, no. 0123456789. Springer Berlin Heidelberg, 2022. doi: 10.1007/s13762-022-04234-4.

[15] L. Magadán, F. J. Suárez, J. C. Granda, and D. F. García, "Low-cost real-time monitoring of electric motors for the Industry 4.0," *Procedia Manuf.*, vol. 42, no. 2019, pp. 393–398, 2020, doi: 10.1016/j.promfg.2020.02.057.

[16] Y. Liu and Z. Ping, "Research on Brand Illustration Innovative Design Modeling Based on Industry 4.0," *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: 10.1155/2022/7475362.

[17] K. P. Paranitharan, G. Ebenezer, V. Balaji, M. Adham Khan, and T. Ramesh Babu, "Application of industry 4.0 technology in containing Covid-19 spread and its challenges," *Mater. Today Proc.*, vol. 68, pp. 1225–1232, 2022, doi: 10.1016/j.matpr.2022.06.009.

[18] L. Da Xu, "Emerging Enabling Technologies for Industry 4.0 and Beyond," *Inf. Syst. Front.*, no. September 2021, 2022, doi: 10.1007/s10796-021-10213-w.

[19] W. Jiang, "A Machine Vision Anomaly Detection System to Industry 4.0 Based on Variational Fuzzy Autoencoder," *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: 10.1155/2022/1945507.

[20] J. Dalzochio *et al.*, "Machine learning and reasoning for predictive maintenance in Industry 4.0: Current status and challenges," *Comput. Ind.*, vol. 123, p. 103298, 2020, doi: 10.1016/j.compind.2020.103298.

[21] C. Maheswari, E. B. Priyanka, S. Thangavel, S. V. R. Vignesh, and C. Poongodi, "Multiple regression analysis for the prediction of extraction efficiency in mining industry with industrial IoT," *Prod. Eng.*, vol. 14, no. 4, pp. 457–471, 2020, doi: 10.1007/s11740-020-00970-z.

[22] M. Sujatha *et al.*, "IoT and Machine Learning-Based Smart Automation System for Industry 4.0 Using Robotics and Sensors," *J. Nanomater.*, vol. 2022, 2022, doi: 10.1155/2022/6807585.

[23] D. Mourtzis, N. Papakostas, and S. Makris, "Complexity in industry 4.0 systems and networks," *Complexity*, vol. 2019, 2019, doi: 10.1155/2019/7817046.