The object of the study is the fault detection process in critical rotating machinery, specifically steam turbines and compressors, operating within a petrochemical production environment. Traditional fault detection methods, though proven and cost-effective, struggle to address modern industrial challenges – such as the increasing complexity of sensor data, class imbalance in failure records, and the need for real-time interpretability. Recent advancements in deep learning offer promising solutions to these limitations. This study proposes an integrated framework that combines Wasserstein Generative Adversarial Network (WGAN) for data balancing and TabNet, an interpretable deep learning model optimized for tabular sensor data. The goal is to enhance the accuracy and interpretability of fault detection under imbalanced, high-dimensional industrial datasets. Using historical data from a petrochemical plant (2015–2024), the WGAN-TabNet model demonstrated superior performance compared to traditional classifiers (Logistic Regression, SVM, XGBoost), achieving an accuracy of 96.01%, precision of 93.25%, recall of 93.14%, F1-score of 93.20%, and AUC score of 93.13%. The interpretability provided by combination of TabNet and SHAP analysis further identified key operational variables influencing failure such as oil temperature and gas flow rate, offering actionable insights for predictive maintenance. The results underscore that integrating deep learning with robust data balancing significantly improves fault detection where traditional methods fall short, supporting practical implementation in modern predictive maintenance systems

Keywords: rotating machinery, fault detection, deep learning, WGAN, TabNet, SHAP, predictive maintenance

UDC: 62-5:004.89:66.013
DOI: 10.15587/1729-4061.2025.332597

FAULT DETECTION OF ROTATING MACHINERY IN THE PETROCHEMICAL INDUSTRY USING A DEEP LEARNING BASED APPROACH: TABNET – WGAN

Muhammad Ikhsan Anshori
Corresponding author
Bachelor of Engineering (Electrical), Master of Engineering (Industrial), Professional Engineer*
E-mail: m.ikhsan.anshori@gmail.com
Arian Dhini
Doctor, Bachelor of Engineering, Master of Engineering, Professional Engineer*
*Department of Industrial Engineering
Universitas Indonesia
Kampus Baru UI str., Pondok Cina, Depok, West Java, Indonesia, 16424

Received 24.03.2025
Received in revised form 16.05.2025
Accepted 11.06.2025
Published 27.06.2025

How to Cite: *Anshori, M. I., Dhini, A. (2025). Fault detection of rotating machinery in the petrochemical industry using a deep learning based approach: TabNet – WGAN. Eastern-European Journal of Enterprise Technologies, 3 (1 (135)), 90–99. https://doi.org/10.15587/1729-4061.2025.332597*

1. Introduction

The reliability of rotating equipment such as steam turbines and compressors is vital for ensuring operational continuity and safety in petrochemical production environments. These machines are central to processes involving fluid compression and energy transformation, yet their complex operational dynamics and prolonged usage make them highly vulnerable to unexpected failures. Such failures, although infrequent, can result in substantial unplanned downtime, pose safety hazards, and lead to considerable financial losses due to interrupted production [1]. These challenges are further compounded by the increasing demands for efficiency, real-time monitoring, and sustainability in modern industrial settings.

A steam turbine operates based on the principle of converting thermal energy from steam into mechanical energy. High-pressure steam is directed onto the turbine blades, causing the rotor to spin. The resulting rotation is transferred via a connected shaft to perform mechanical work [2]. Compressors, on the other hand, increase the pressure of a gas by reducing its volume, using rotating components such as blades or pistons. Low-pressure gas is drawn in, compressed adiabatically, and exits at a higher pressure [3]. Although steam turbines and compressors perform distinct functions, they share common characteristics such as rotational motion, critical speed limitations, and vulnerability to mechanical issues such as imbalance, misalignment, and vibration [4].

Traditional maintenance strategies, such as reactive and preventive maintenance, have proven insufficient in addressing these challenges. Reactive maintenance is inefficient due to its post-failure nature, while preventive maintenance risks over-maintenance or unforeseen breakdowns. Predictive maintenance, which leverages historical data to forecast equipment failures, offers a promising alternative, enabling fault detection and minimizing unplanned downtime [5]. However, predictive maintenance for failure detection in industrial settings has relied on manual inspection and rule-based systems, which are limited in handling the high volume, velocity, and variability of modern sensor data [6].

In modern petrochemical operations, the adoption of Industry 4.0 technologies has driven significant growth in real-time process monitoring through advanced sensor networks. While these developments offer new opportunities for predictive maintenance, they also introduce substantial challenges. Industrial rotating machinery now operates within highly dynamic environments characterized by vast and

complex sensor data, strict efficiency targets, and heightened safety and environmental regulations [7].

The previous fault detection techniques, including manual inspections, rule-based monitoring, and classical machine learning, often prove insufficient in this context [8, 9]. These methods typically struggle with severe class imbalance, nonlinear variable interactions, and the need for interpretable outputs that can guide operational decisions. Moreover, as industry expectations for predictive accuracy and transparency continue to rise, there is a growing demand for fault detection models that can effectively handle complex, imbalanced industrial datasets while providing actionable insights to maintenance practitioners [9].

Consequently, scientific research on advanced, interpretable deep learning frameworks supported by robust data balancing techniques is both timely and essential. These studies directly address modern challenges in predictive maintenance and has the potential to deliver significant practical benefits: reducing unplanned downtime, improving operational reliability, optimizing maintenance planning, and enhancing plant safety [10].

2. Literature review and problem statement

Recent years have seen significant progress in the development of data-driven fault detection techniques for industrial rotating machinery. Numerous studies have explored various machine learning and deep learning approaches aimed at improving predictive maintenance accuracy and reliability. The literature in this field generally falls into three key areas: the design of advanced classification algorithms for fault detection, strategies to address class imbalance in fault datasets, and techniques to improve model interpretability to support practical maintenance decision-making.

Traditional algorithms such as logistic regression and support vector machines (SVM) offer ease of implementation and interpretability but tend to underperform in capturing nonlinear and complex feature interactions commonly present in high-dimensional industrial data [11]. XGBoost, a more advanced gradient boosting algorithm, has shown improved predictive performance and robustness in handling structured data, and is widely used in fault classification tasks [12]. However, despite their effectiveness, these classifiers still face significant challenges when applied to highly imbalanced datasets where failure instances are significantly underrepresented, which is leading to biased models that favor the majority (non-failure) class and exhibit poor sensitivity to minority-class patterns [13].

To address the class imbalance problem, resampling strategies such as undersampling and Synthetic Minority Over-sampling Technique (SMOTE) are often adopted [14]. Nonetheless, these approaches have inherent drawbacks, such as loss of information or the generation of synthetic samples that poorly represent real-world failure conditions [15]. In recent developments, Generative Adversarial Networks (GAN) has emerged as a powerful tool to create synthetic data that better reflects the underlying data distribution [16]. However, standard GANs frequently encounter training instabilities and mode collapse [17]. The Wasserstein Generative Adversarial Network (WGAN), a variant of GAN, has been proposed to improve convergence behavior and generate more diverse and reliable synthetic data, particularly for underrepresented fault cases [7].

At the same time, deep learning models specifically designed for tabular data have gained interest. TabNet is one such model that integrates attention-based feature selection mechanisms to process structured data directly, without requiring extensive manual feature engineering. In contrast to traditional neural networks like CNNs or RNNs, which are better suited for images or sequential signals, TabNet is tailored to industrial datasets that are often tabular and multidimensional [18]. Furthermore, TabNet provides enhanced interpretability through its compatibility with SHAP (Shapley Additive Explanations), making it appropriate for applications where model transparency is crucial [19].

Despite these advancements, limited research has explored the integration of TabNet with generative oversampling techniques like WGAN for fault detection applications in the petrochemical sector. Most prior studies have either focused solely on improving classification algorithms or on addressing class imbalance independently [20]. This gap highlights the need for a unified framework that combines interpretable deep learning classifiers with robust synthetic data generation techniques to improve detection accuracy in real-world, imbalanced industrial datasets.

All this allows to assert that it is expedient to conduct a study on the development of an integrated WGAN-TabNet-based framework that combines accurate detection of rare fault events with interpretable insights, thereby advancing predictive maintenance capabilities for rotating machinery in the petrochemical industry.

3. The aim and objectives of the study

The aim of the study is to development of an integrated approach that combines robust data balancing with interpretable modeling, tailored to the operational demands of the petrochemical industry.

To achieve this aim, the following objectives are accomplished:

– to assess the performance limitations of the TabNet classifier when trained directly on imbalanced fault data without any data balancing strategy;

– to develop and evaluate a WGAN-based synthetic data generation process for balancing minority fault classes and to integrate it with TabNet to improve rare event detection;

– to compare the performance of the WGAN-TabNet model against benchmark classifiers (Logistic Regression, SVM, XGBoost) in order to validate its robustness and generalization;

– to identify and interpret key sensor variables that contribute most to fault prediction by applying SHAP (SHapley Additive exPlanations) to the TabNet classifier outputs.

4. Materials and methods

4. 1. Object and hypothesis of the study

The object of the study is the fault detection process in critical rotating machinery, specifically steam turbines and compressors, operating within a petrochemical production environment. The study focuses on the utilization of high-dimensional and class-imbalanced industrial sensor data collected from these machines during typical operational cycles.

The main hypothesis of the study is that integrating Wasserstein Generative Adversarial Network (WGAN) for synthetic data balancing with TabNet for interpretable classification can significantly enhance the accuracy of detecting rare fault events in rotating machinery, while also providing transparent and actionable insights into the factors contributing to such failures.

Several assumptions underlie the study. It is assumed that the historical sensor data collected from the target rotating equipment accurately reflects both normal and faulty operating conditions typical of the petrochemical industry. Furthermore, it is assumed that the synthetic failure data generated by WGAN sufficiently preserves the statistical characteristics of actual fault data, thereby avoiding distortions that could compromise model training. The TabNet architecture is presumed suitable for learning complex relationships within the high-dimensional tabular sensor data used for fault detection. Additionally, the benchmark classifiers selected for comparison (Logistic Regression, Support Vector Machine, and XGBoost) are assumed to be adequately optimized to serve as fair baselines.

To manage the scope of the study, certain simplifications have been adopted. The research focuses exclusively on fault detection, treated as a binary classification problem (failure versus non-failure), and does not extend to fault diagnosis or severity estimation.

The evaluation is performed on an offline dataset rather than a real-time streaming environment. Moreover, the study is based on data from a single petrochemical plant unit, and cross-validation across multiple plants or equipment types is reserved for future work. Finally, potential external influences such as operator interventions and maintenance activities are not explicitly modeled within the current analytical framework.

4. 2. Conceptual framework

This study follows a structured methodology for failure detection in rotating machinery by addressing the challenges of high-dimensional and imbalanced data. The proposed model integrates WGAN for synthetic minority oversampling and TabNet as the core classifier. As shown in Fig. 1, the dataset used in this study was obtained from PT Pupuk Kujang, a state-owned petrochemical company in Indonesia, consisting of historical operational data collected between 2015 and 2024 from unit steam turbine and syngas compressor A-103-J. It includes 147 process parameters such as vibration, pressure, temperature, flow, analyzer outputs, and speed measurements. They were used as inputs for data-driven fault detection model. For classification, the primary model used was TabNet. To structure the evaluation of the proposed methodology, three experimental paths were designed.

The first path represents the baseline model, where classification is performed directly on the pre-processed data without balancing.

The second path introduces WGAN after preprocessing to generate synthetic failure samples and address the class imbalance. This path also evaluates the effect of balancing training data on model performance.

Lastly, the third path explores the same hybrid preprocessing flow but replaces TabNet with traditional classifiers, namely Logistic Regression, Support Vector Machine (SVM), and XGBoost, to assess the comparative advantage of using TabNet.

Furthermore, to assess model interpretability, SHAP (SHapley Additive exPlanations) was applied to TabNet's output to determine the most influential features in predicting failure. The combined approach of preprocessing, synthetic balancing, and deep learning classification aims to provide a robust and interpretable solution for predictive maintenance in complex industrial settings.

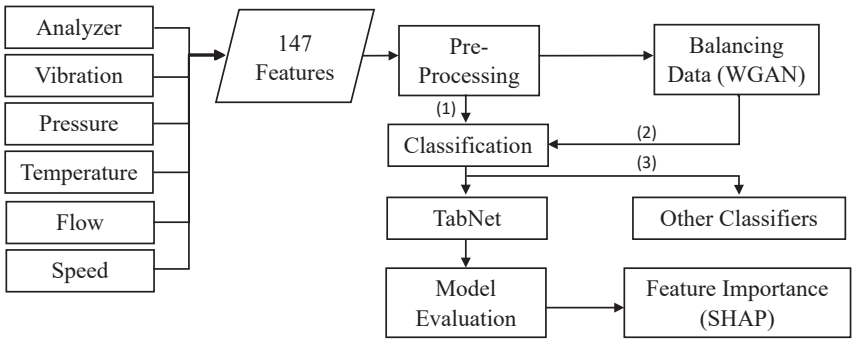

Fig. 1. Conceptual framework

4. 3. Data pre-processing

The pre-processing stage was conducted using Python programming language and involved several systematic activities to ensure data consistency and readiness for further analysis. The first step was data cleaning, which focused on identifying and removing duplicate records based on the tag number of the parameters. Additionally, rows containing missing values were entirely excluded from the dataset. This decision was taken to avoid introducing biases or inaccuracies, especially considering that missing values often appear across multiple process parameters recorded at the same timestamp. As a result of this initial cleaning process, the number of features (tag numbers) was reduced from 147 to 140, while the number of records decreased from 17,505 to 17,477. As seen in Fig. 2, several variables exhibit significantly different scales and ranges, with some features displaying values reaching up to 20,000 while others remain below 100. This indicates a high degree of heterogeneity in the data, particularly because the data has not yet undergone normalization or standardization processes.

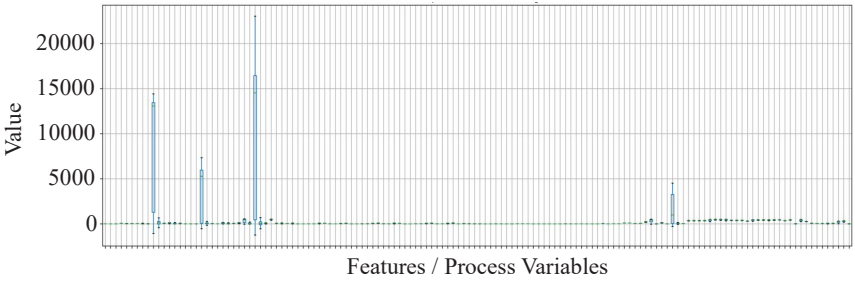

Fig. 2. Boxplot after cleaning data

Subsequently, standardization was applied to all numerical features using Z-score normalization. This step was essential because, without standardization, features with larger magnitudes could dominate the learning process, potentially leading to sub-optimal model performance and convergence issues. Z-score normalization transforms the original data by rescaling it to have a mean of zero and a standard deviation of one, ensuring that each feature contributes equally during model training. The standardization formula used is as follows

$$Z = \frac{X - \mu}{\sigma}, \quad (1)$$

where X – the original value, μ – the mean and σ – the standard deviation of the respective feature. Fig. 3 presents the boxplot of all features after standardization, showing that most variables now fall within a standardized range of approximately −3 to +3, with interquartile ranges centered around zero. This indicates that the standardization process has successfully normalized the scales across features, allowing for meaningful comparison and analysis. However, the presence of extended whiskers and some outliers in several variables suggests that certain operational parameters still exhibit inherent variability, which may carry valuable information for anomaly detection or predictive maintenance.

Following data cleaning and standardization, the dataset was divided into training and testing subsets using an 80:20 ratio. The training set was utilized to develop machine learning models, while the testing set was reserved for performance evaluation. After completing data splitting steps, the training set consisted of 13,981 records, comprising 11,429 normal and 2,552 failure instances. Meanwhile, the testing set contained 3,496 records, including 2,878 normal and 618 failure instances. This final distribution reflects the imbalanced nature of the dataset, highlighting the necessity of appropriate data balancing strategies during the modelling process.

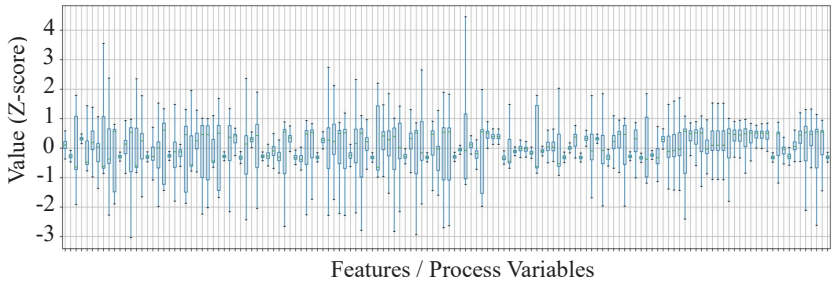

Fig. 3. Boxplot of standardized data

4. 4. Data balancing strategies

To address the class imbalance issue inherent in the failure detection dataset where the minority class (failures) represented by 2552 failure instances or less than 19% of total records, this study employed the Wasserstein Generative Adversarial Network (WGAN) to generate synthetic fault data. The WGAN architecture comprises two neural networks: a generator G and a critic D, which are trained simultaneously in a minimax optimization framework. Unlike classical GANs that minimize the Jensen-Shannon divergence, WGAN minimizes the Earth Mover's distance (also known as Wasserstein-1 distance), which is more stable and informative in gradient propagation. The loss function of WGAN can be defined as

$$\min_G \max_{D \in \mathcal{D}} E_{x \sim P_r}\left[D(x)\right] - E_{z \sim P_z}\left[D(G(z))\right], \quad (2)$$

where P_r represents the real data distribution, P_z denotes the distribution of noise variables, and D – the set of 1-Lipschitz functions enforced via gradient penalty. The training process involved alternating updates between the critic and the generator, with the critic trained five times more frequently per iteration to ensure the accurate estimation of Wasserstein distance.

The WGAN model was trained using the Adam optimizer and tuned with the following settings shown in Table 1.

Table 1

Hyperparameter configuration

Hyperparameter	Value	Description
Latent Dimension	32	Dimension of the random noise vector used as input to the generator. This value affects the generator's capacity to learn complex patterns
Batch Size	64	The number of samples processed in each training iteration. This value influences both the stability and speed of the training process
Epochs	5000	The total number of training iterations performed to achieve convergence
n_critic	5	The number of training steps for the critic per training step of the generator
Learning Rate	1e-4	The learning rate used for the Adam optimizer applied to both the generator and the critic
Gradient Penalty	10	Coefficient used to enforce the Lipschitz continuity constraint via gradient penalty

This configuration allowed WGAN to generate synthetic failure instances that closely mirrored the statistical characteristics of real fault data. The augmented dataset was subsequently used to train the TabNet classifier, thereby improving the model's performance in detecting rare fault conditions.

4. 5. TabNet classifier

Tabular data is prevalent in many industries, often characterized by structured and heterogeneous features. Conventional models, such as decision trees and gradient boosting machines, perform well but lack inherent deep feature representation. Conversely, deep learning models frequently struggle with interpretability and efficiency in handling tabular data.

TabNet addresses these challenges through an attentive, sequential processing framework that enables effective learning of data-driven representations with high interpretability. This feature-wise attention allows TabNet to perform implicit feature selection during training, emphasizing the most relevant inputs while suppressing less informative ones. The architecture of TabNet consists of multiple components as shown in Fig. 4.

The model begins by transforming the input features through a shared feature transformer, which processes input features through fully connected layers and applies ReLU activation to introduce non-linearity. In each decision step, the Attentive Transformer generates sparse feature masks, which select relevant features for the subsequent step, encouraging sparsity via the Sparsemax activation function. The mask $M^{[l]}$ at decision step l is computed as

$$MM^{[l]} = \text{Sparsemax}\left(P^{[l-1]} \odot h^{[l-1]}\right), \quad (3)$$

where $\odot$ denotes element-wise multiplication, and Sparsemax promotes sparsity by selecting a small subset of important features. The decision outputs at each step are then aggregated for the final prediction. The output of the model is the average of the decision steps for classification tasks or the sum for regression tasks

$$\hat{y}=\frac{1}{L}\sum_{l=1}^{L}d^{[l]}, \quad (4)$$

where $d^{[l]}$ – the decision output at step l and is the total number of decision steps. This sequential decision-making process ensures that the model progressively refines its understanding of the data, making TabNet an efficient and interpretable approach for tabular data analysis.

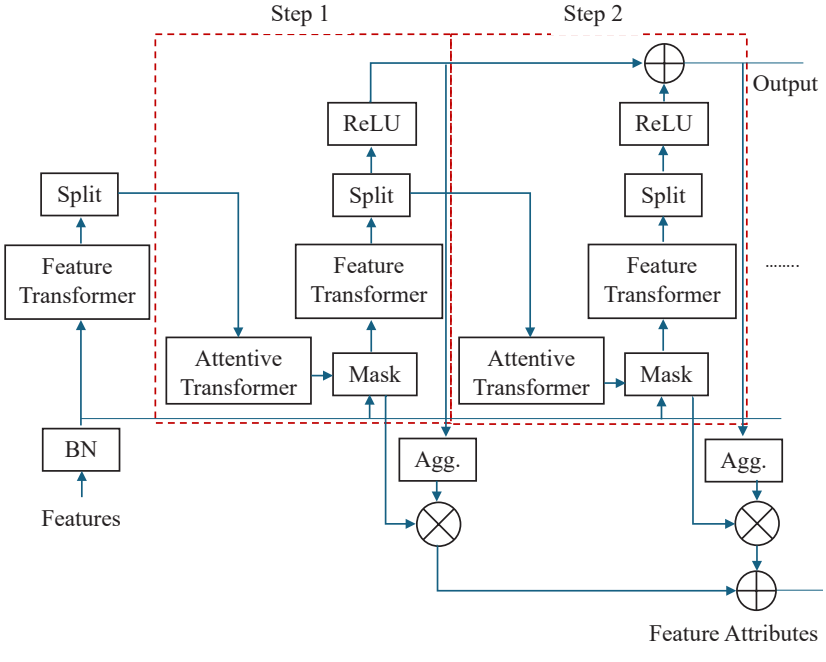

Fig. 4. Architecture diagram of TabNet

In this study, TabNet was configured using a series of grid search experiments to determine appropriate hyperparameter values. The hyperparameters used for TabNet training are shown in Table 2.

The selected hyperparameter values in Table 2 reflect a trade-off between learning efficiency and model stability. A relatively small learning rate (0.005) combined with scheduled decay every 50 epochs helps prevent overshooting and encourages convergence.

The use of StepLR and gamma decay ensures smooth training progress, while the verbose setting facilitates monitoring. This configuration was found to be effective in maximizing the TabNet model's ability to learn complex failure patterns without overfitting.

Table 2

Hyperparameter configuration

Hyperparameter	Value	Description
Learning rate	0.005	Controls the rate of weight updates during each training iteration
Scheduler function	StepLR	Adjusts the learning rate gradually throughout the training process
Step Size	50	The learning rate is reduced every 50 epochs
Gamma	0.9	Multiplicative factor for the learning rate at each adjustment step
Verbose	10	Determines the logging frequency during training (every 10 steps)

4. 6. Feature importance

To complement the predictive performance of the TabNet classifier with a transparent decision-making process, this study incorporated SHAP (SHapley Additive exPlanations) for post-hoc interpretability analysis. SHAP is a unified framework derived from cooperative game theory that attributes the contribution of each input feature to the model's prediction, providing both global and local interpretability.

After the final TabNet model was trained on the WGAN-balanced dataset, SHAP values were computed to quantify the importance of each process parameter in the fault classification task. The SHAP values ϕ_j for feature f_j are computed by averaging its marginal contributions across all possible feature coalitions

$$\phi_j=\sum_{S\subseteq F\setminus\{j\}}\frac{|S|!)|F|-|S|-1)!}{|F|!}\times$$
$$\times\left[f_{S\cup\{j\}}\left(x_{S\cup\{j\}}\right)-f_S\left(x_S\right)\right], \quad (5)$$

where F – the set of all features, S represents subsets of features excluding feature f, and $f_s(x_s)$ denotes the expected model prediction when using the subset S. This analysis enabled the identification of variables that had the most significant influence on the model's output, thus allowing for engineering insights into potential root causes of equipment failure.

5. Results of research on fault detection model scenarios

5. 1. Baseline model performance (without data balancing)

In the initial experimental scheme, the TabNet classifier was trained using the original imbalanced dataset without applying any resampling or data balancing technique. The purpose of this baseline scenario was to observe how the classifier performs when exposed to the raw class distribution, which is heavily skewed toward normal operational conditions.

The evaluation metrics for both training and testing data are summarized in Fig. 5. On the training dataset, the model achieved an accuracy of 0.9635, precision of 0.9006, recall of 0.8998, F1-score of 0.9002, and AUC of 0.9423. However, performance declined slightly on the testing dataset, with an accuracy of 0.9622, precision of 0.8985, recall of 0.8889, F1-score of 0.8937, and AUC of 0.9299.

These results indicate that although TabNet demonstrated relatively strong performance in terms of overall accuracy and AUC, its ability to consistently identify minority class instances, reflected in the lower recall and F1-score, was still limited. This outcome highlights the sensitivity of deep learning classifiers to imbalanced class distributions, where the majority class tends to dominate learning behavior.

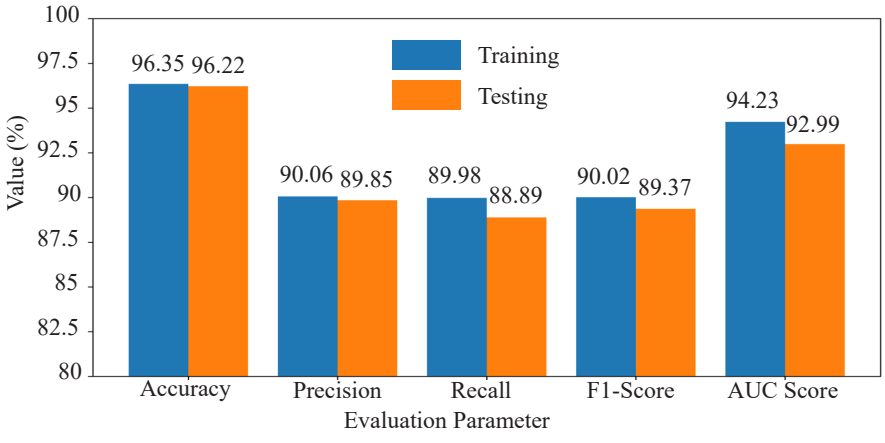

Fig. 5. Baseline model

5. 2. Performance of the WGAN-TabNet integration

In the second experimental scheme, the dataset was augmented using the Wasserstein Generative Adversarial Network (WGAN) to synthetically balance the failure class before being used to train the TabNet classifier. This integration was intended to mitigate the negative effects of class imbalance observed in the baseline scenario and to improve the model's ability to detect rare failure instances.

As shown in Fig. 6, the WGAN-TabNet model achieved a training accuracy of 0.9706, with precision, recall, F1-score, and AUC all equal to 0.9706. These results indicate a strong learning capability with no apparent signs of overfitting on the training set. When evaluated on the test dataset, the model maintained high generalization performance, achieving 0.9601 accuracy, 0.9325 precision, 0.9314 recall, 0.932 F1-score, and 0.9313 AUC score.

Compared to the baseline model, the integration of WGAN contributed significantly to the improvement of recall and F1-score, which are critical metrics in predictive maintenance tasks. The high recall suggests that the model became more sensitive to detecting failure events, while the consistent AUC demonstrates balanced classification capability across various thresholds.

5. 3. Performance of benchmark classifiers

To assess the performance of the proposed WGAN-TabNet model, three commonly used classification algorithms, namely Logistic Regression, Support Vector Machine (SVM), and Extreme Gradient Boosting (XGBoost), were selected as benchmark models. Each classifier was trained using the same WGAN-balanced dataset to ensure a fair comparison across methods. The evaluation metrics for each model on both training and testing datasets are presented in Fig. 7.

On the training data, logistic regression achieved an accuracy of 0.9314, with precision, recall, F1-score, and AUC all at 0.9314. However, performance slightly declined on the testing dataset, with an accuracy of 0.9468, precision of 0.9008, recall of 0.9241, F1-score of 0.9119, and AUC of 0.9241.

The SVM model showed improved results compared to logistic regression. On the training dataset, SVM yielded 0.9482 accuracy, 0.9494 precision, 0.9482 recall, 0.9481 F1-score, and 0.9482 AUC. These values were relatively consistent on the testing dataset, where SVM achieved 0.959 accuracy, 0.9296 precision, 0.9307 recall, 0.9302 F1-score, and 0.9307 AUC.

Among the three benchmarks, XGBoost exhibited the strongest performance on the training dataset, with perfect alignment across all metrics at 0.9789. This indicates that XGBoost is highly well-fitted to the training data. On the test data, however, its accuracy dropped to 0.9598, with precision of 0.9312, recall of 0.9316, F1-score of 0.9314, and AUC of 0.9316, still strong, but slightly less consistent than the training set. Despite the high results from XGBoost and SVM, both models demonstrated a tendency to overfit or exhibit reduced generalization compared to TabNet.

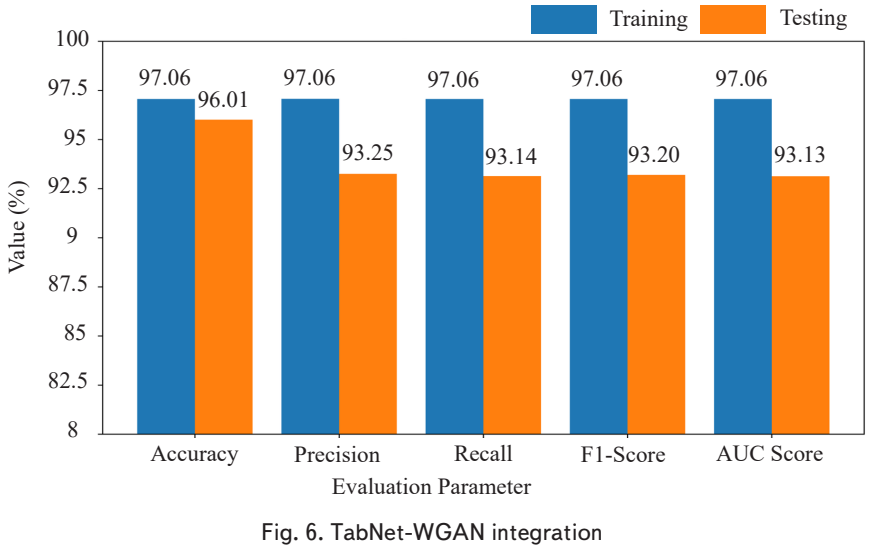

Fig. 6. TabNet-WGAN integration

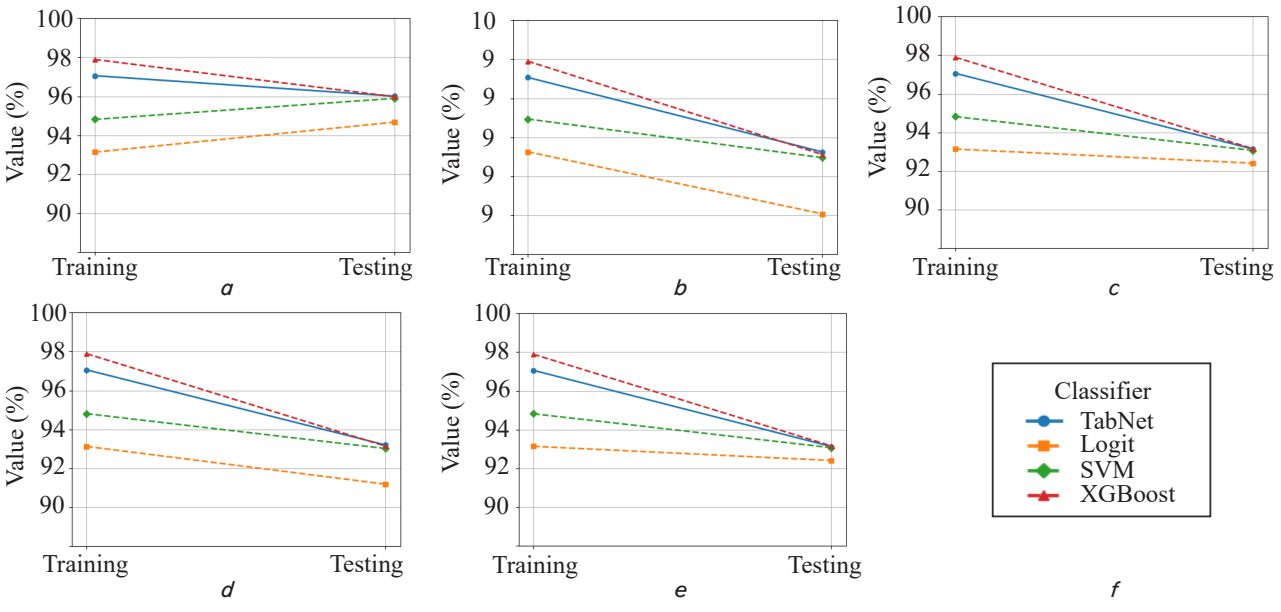

Fig. 7. Comparison between classifiers: *a* — accuracy; *b* — precision; *c* —recall; *d* — F1-score; *e* — AUC score; *f* — legend

5. 4. Feature importance from SHapley Additive exPlanations

To further enhance the interpretability of the TabNet model, SHAP (SHapley Additive exPlanations) analysis was employed to evaluate the contribution of each feature to the model's predictions. SHAP values provide insights into how each feature influences the model's output, both globally and locally. This analysis is crucial for understanding the key factors driving fault detection predictions in the rotating machinery dataset.

The SHAP summary plot shown in Fig. 8 provides an overview of feature importance across all samples. Each point in the plot represents a SHAP value for a feature and corresponds to an individual instance in the dataset. The horizontal axis shows the impact of each feature on the model output, with features positioned according to their overall influence. From the summary plot, it is evident that feature TI5011A.PV and feature FIC1008.PV have the largest impact on the model's predictions, both displaying significant variation in SHAP values. Features with high SHAP values (in red) indicate that the corresponding feature values contribute positively to the model's prediction, whereas low SHAP values (in blue) have the opposite effect.

The SHAP waterfall plot (Fig. 9) illustrates how individual features influence a specific prediction. In this case, it is possible to observe that Feature 65 has the most significant positive impact on the prediction of fault occurrence, increasing the probability of failure substantially. The cumulative effect of features is clearly shown, allowing for a detailed understanding of how each feature value pushes the prediction toward failure or non-failure. The contribution of Feature 22 and Feature 2 further supports the prediction, while Feature 64 contributes negatively, lowering the predicted probability of failure.

The SHAP bar chart (Fig. 10) aggregates the mean absolute SHAP values for each feature, providing a ranking of their overall importance across all instances. Feature 65 stands out as the most influential feature, with a SHAP value of +0.15. Following it are Feature 22 (+0.12) and Feature 13 (+0.03), which also contribute significantly to the fault prediction. The chart also highlights that a large number of other features, such as Feature 7 and Feature 91, have a minor contribution to the final prediction, with SHAP values close to zero.

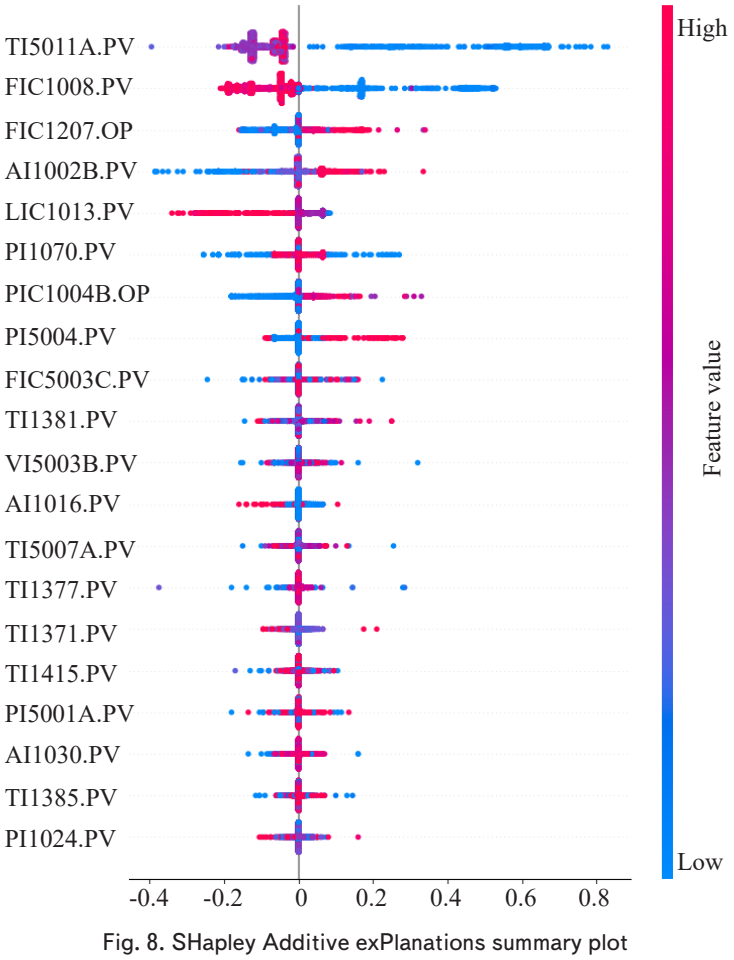

Fig. 8. SHapley Additive exPlanations summary plot

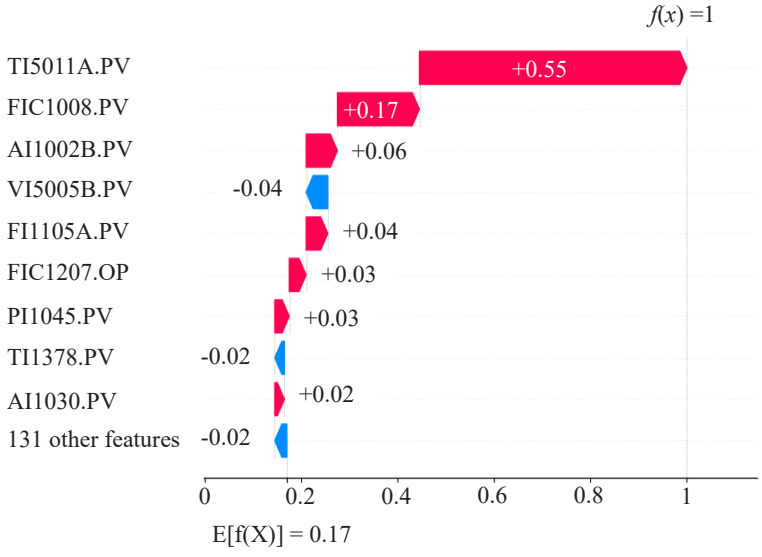

Fig. 9. SHapley Additive exPlanations waterfall plot

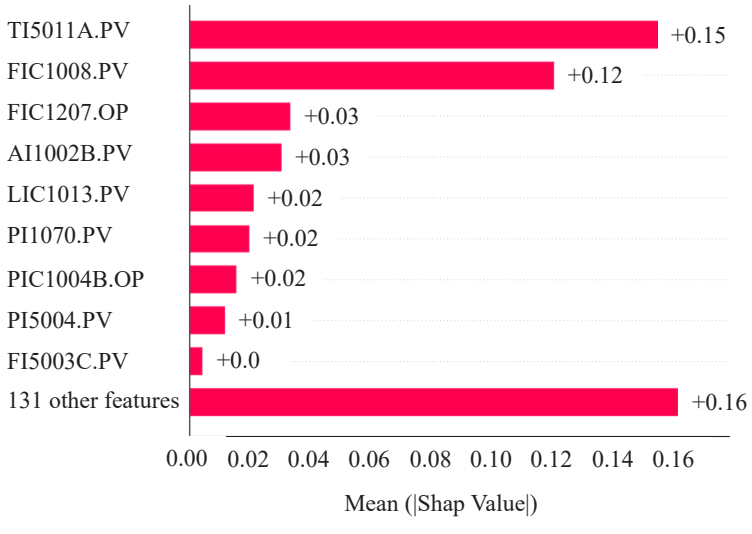

Fig. 10. SHapley Additive exPlanations bar chart

The combined insights from the SHAP summary plot, waterfall plot, and bar chart consistently highlight that a few critical features, TI5011A.PV (oil temperature) and FIC1008.PV (inlet gas flow) play a dominant role in fault prediction. These features appeared repeatedly with high SHAP values and were responsible for significant shifts in prediction probabilities toward failure events.

6. Discussion of results from fault detection model scenarios

The integration of Wasserstein Generative Adversarial Network (WGAN) and TabNet demonstrated significant improvements in fault detection performance for rotating machinery under class-imbalanced conditions. As illustrated in Fig. 5 and Fig. 6, training TabNet directly on the imbalanced dataset led to high accuracy (0.9622) but relatively lower recall (0.8889) and F1-score (0.8937), reflecting the model's bias toward the majority class. This outcome aligns with previous findings in literature [4, 13, 19], where standard classifiers tend to underperform in rare event detection due to skewed class distributions.

In contrast, the WGAN-TabNet integration effectively mitigated this imbalance, achieving substantial gains in recall (0.9314) and F1-score (0.932), while maintaining balanced accuracy and AUC (Fig. 6). This confirms that the synthetic samples generated by WGAN successfully preserved the statistical structure of minority-class data, as supported by previous studies that reported similar benefits of GAN-based balancing in fault detection contexts [7, 18].

Benchmark comparison (Fig. 7) shows that while XGBoost and SVM achieved competitive performance, TabNet-WGAN provided better generalization and interpretability. Although XGBoost yielded perfect training results (0.9789), it exhibited slight overfitting on test data (0.9598 accuracy). Moreover, unlike TabNet, these models lack built-in feature selection and transparency mechanisms. This supports the conclusions from earlier investigation emphasizing the advantage of attention-based deep learning models in interpretable tabular learning [5, 18].

SHAP-based interpretability (Fig. 8–10) further reinforces the model's credibility. Feature TI5011A.PV (oil temperature) and feature FIC1008.PV (inlet gas flow) were identified as top contributors, consistent with known failure precursors in rotating equipment [8–10]. The agreement between model insights and engineering knowledge supports the model's applicability in real-world maintenance planning.

Overall, this study validates the feasibility of combining WGAN and TabNet for fault detection in imbalanced industrial data, providing both predictive performance and transparency. This work contributes to the body of research on interpretable deep learning for condition monitoring and aligns with the industrial demands for reliability, explainability, and actionable insight.

Despite its strengths, this study has certain limitations. First, the dataset originates from a single petrochemical plant, limiting generalizability to other industrial settings. Second, the framework was tested in an offline batch learning context, without real-time data streaming, which restricts immediate deployment in online monitoring systems.

Among the disadvantages, WGAN training is computationally intensive and sensitive to hyperparameters. Additionally, TabNet's performance is reliant on careful tuning and may not scale well for extremely sparse datasets or environments with limited computational resources.

For future development, several directions are recommended to enhance the applicability and robustness of the proposed fault detection framework. First, validation should be extended to various types of rotating machinery and across multiple industrial sites to improve the generalizability of the model. This would ensure that the approach is not limited to a single plant context. Second, integrating the model into real-time monitoring systems and enabling online learning capabilities would allow continuous adaptation to new data and changing operational conditions, supporting more dynamic predictive maintenance. Third, exploring alternative or lightweight data balancing methods with lower computational

complexity, such as variational autoencoders or improved synthetic sampling could address the limitations posed by the training demands of WGAN. Additionally, the model's efficiency may be optimized by removing features with low SHAP value contributions, allowing the system to run faster and more resource-efficient without sacrificing accuracy. Finally, embedding explainability mechanisms directly into maintenance dashboards would strengthen operator trust and enable timely, informed decision-making in industrial environments.

7. Conclusions

1. The experiment on the baseline model using the TabNet classifier on the original imbalanced dataset demonstrated significant challenges in detecting minority class failures. The model achieved an accuracy of 96.22%, precision of 89.85%, recall of 88.89%, F1-score of 89.37%, and AUC of 92.99% on the testing set. Although the model exhibited good overall performance, its recall and F1-score were relatively low, highlighting the difficulty of accurately detecting rare fault events without addressing the class imbalance issue.

2. The integration of WGAN for synthetic data balancing improved the model's ability to detect rare fault events significantly. The WGAN-TabNet model achieved 96.01% accuracy, 93.25% precision, 93.14% recall, 93.20% F1-score, and 93.13% AUC on the testing dataset. These results demonstrate that WGAN effectively mitigates the bias towards the majority class, enhancing recall and F1-score while maintaining balanced accuracy and AUC. This improvement confirms the effectiveness of WGAN in handling class imbalance in fault detection tasks.

3. Performance benchmarking against traditional classifiers (Logistic Regression, SVM, XGBoost) further validated the superiority of the WGAN-TabNet integration. On the test set, Logistic Regression achieved an F1-score of 91.19%, SVM achieved 92.96%, and XGBoost scored 93.14%, all lower than the 93.20% achieved by WGAN-TabNet. Moreover, WGAN-TabNet outperformed these classifiers in terms of AUC, achieving 93.13% compared to XGBoost's 93.01%, SVM's 92.57%, and Logistic Regression's 91.96%. These results highlight the robust generalization and improved detection capability of WGAN-TabNet.

4. The analysis of SHAP values allowed the extraction of meaningful operational insights, the feature TI5011A.PV (oil temperature) showed the highest average SHAP value of +0.15, followed by FIC1008.PV (inlet gas flow) with +0.12. These findings highlight the importance of specific operational variables in predicting equipment failures, providing valuable insights for maintenance engineers. The SHAP analysis further reinforces the transparency and interpretability of the WGAN-TabNet model, enabling more informed decision-making for predictive maintenance interventions.

Conflict of interest

The authors declare that they have no conflicts of interest related to this research, whether personal, publication-related, or otherwise, that could influence the research and the results presented in this paper.

Financing

All funding for this research was provided by PT Pupuk Kujang, by sending their employee to pursue a master's degree.

Data availability

Data will be made available on reasonable request.

Use of artificial intelligence

The authors have used artificial intelligence technology within acceptable limits, solely to refine word choices in this manuscript, and it was not used as a tool for analyzing or interpreting test results.

Acknowledgments

The authors would like to express their deepest gratitude to PT Pupuk Kujang Cikampek for their full financial support, as well as for providing facility, data, and the materials to complete this research. Without the substantial support from this company, this research would not have been possible.

References

1. Mobley, R. K. (2002). An Introduction to Predictive Maintenance. Butterworth-Heinemann. https://doi.org/10.1016/b978-0-7506-7531-4.x5000-3
2. Borgnakke, C., Sonntag, R. E. (2013). Fundamentals of Thermodynamics. Wiley, 912.
3. Giampaolo, T. (2010). Compressor Handbook: Principles and Practice. The Fairmont Press, 376.
4. Mobley, R. K. (2001). Plant Engineer's Handbook. Butterworth-Heinemann.
5. Moubray, J. (1997). Reliability-Centered Maintenance. Butterworth-Heinemann.
6. Nunes, P., Santos, J., Rocha, E. (2023). Challenges in predictive maintenance – A review. CIRP Journal of Manufacturing Science and Technology, 40, 53–67. https://doi.org/10.1016/j.cirpj.2022.11.004
7. Zhou, H., Pan, H., Zheng, K., Wu, Z., Xiang, Q. (2025). A novel oversampling method based on Wasserstein CGAN for imbalanced classification. Cybersecurity, 8 (1). https://doi.org/10.1186/s42400-024-00290-0
8. Jardine, A. K. S., Lin, D., Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mechanical Systems and Signal Processing, 20 (7), 1483–1510. https://doi.org/10.1016/j.ymssp.2005.09.012
9. Zhang, W., Yang, D., Wang, H. (2019). Data-Driven Methods for Predictive Maintenance of Industrial Equipment: A Survey. IEEE Systems Journal, 13 (3), 2213–2227. https://doi.org/10.1109/jsyst.2019.2905565
10. Goodfellow, I., Bengio, Y., Courville, A. (2016). Deep Learning. The MIT Press, 800.

11. Liu, R., Yang, B., Zio, E., Chen, X. (2018). Artificial intelligence for fault diagnosis of rotating machinery: A review. Mechanical Systems and Signal Processing, 108, 33–47. https://doi.org/10.1016/j.ymssp.2018.02.016
12. Chen, T., Guestrin, C. (2016). XGBoost. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794. https://doi.org/10.1145/2939672.2939785
13. Tarekegn, A. N., Giacobini, M., Michalak, K. (2021). A review of methods for imbalanced multi-label classification. Pattern Recognition, 118, 107965. https://doi.org/10.1016/j.patcog.2021.107965
14. Chawla, N. V., Bowyer, K. W., Hall, L. O., Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16, 321–357. https://doi.org/10.1613/jair.953
15. Blagus, R., Lusa, L. (2013). SMOTE for high-dimensional class-imbalanced data. BMC Bioinformatics, 14 (1). https://doi.org/10.1186/1471-2105-14-106
16. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S. et al. (2014). Generative Adversarial Networks. Advances in Neural Information Processing Systems (NeurIPS). arXiv. https://doi.org/10.48550/arXiv.1406.2661
17. Arjovsky, M., Chintala, S., Bottou, L. (2017). Wasserstein GAN. International Conference on Machine Learning (ICML). arXiv. https://doi.org/10.48550/arXiv.1701.07875
18. Arik, S. Ö., Pfister, T. (2021). TabNet: Attentive interpretable tabular learning. Proceedings of the AAAI Conference on Artificial Intelligence. arXiv. https://doi.org/10.48550/arXiv.1908.07442
19. Fares, I. A., Abd Elaziz, M. (2025). Explainable TabNet Transformer-based on Google Vizier Optimizer for Anomaly Intrusion Detection System. Knowledge-Based Systems, 316, 113351. https://doi.org/10.1016/j.knosys.2025.113351
20. Fan, J., Yuan, X., Miao, Z., Sun, Z., Mei, X., Zhou, F. (2022). Full Attention Wasserstein GAN With Gradient Normalization for Fault Diagnosis Under Imbalanced Data. IEEE Transactions on Instrumentation and Measurement, 71, 1–16. https://doi.org/10.1109/tim.2022.3190525