Applied Engineering Letters Vol.10, No.3, 171-182 (2025) e-ISSN: 2466-4847

COMPARATIVE ANALYSIS OF MACHINE LEARNING METHODS FOR SOLVING THE PROBLEM OF PREDICTING FAILURES IN GAS TURBINE ENGINES

UDC:621.438:004.85
Original scientific paper https://doi.org/10.46793/aeletters.2025.10.3.5

Maria Lapina[1], Mikhail Kondrashov[2], Mikhail Babenko[1], Feroz Shaik[3], B. Deepanraj[3*]

[1]North-Caucasus Federal University, 355017, Stavropol, Russia
[2]MIREA – Russian Technological University, Russian Federation, 119454, Moscow, Russia
[3]Prince Mohammad Bin Fahd University, Al-Khobar, 31952, Saudi Arabia

Abstract:
Gas turbine energy technologies are one of the most important components of the modern and advanced energy industry. An important task is to ensure the uninterrupted operation of the equipment in a given period; therefore, monitoring and diagnostics of the technical condition of the equipment continue to play an important role in ensuring the quality of the gas turbine engine. The article examines the work on equipment diagnostics using machine learning. It discusses various solutions for combining machine-learning methods and dealing with unbalanced data to solve the problem of predicting the failure of gas turbine equipment on a dataset that has the above disadvantages. There is a review of the solutions and methods under consideration to deal with the problems of the dataset. At the end, the authors provide a comparative table of the results of the application of the considered solutions based on the quality metrics of the Recall, Precision, F1-score classification, and PR-AUC and ROC-AUC curves.

ARTICLE HISTORY
Received: 5 June 2025
Revised: 30 August 2025
Accepted: 13 September 2025
Published: 30 September 2025

KEYWORDS
Machine learning, equipment failure prediction, gas turbine engine, gas turbine power plant, data imbalance, fuzzy logic, SMOTE, Tomek Links

1. INTRODUCTION

Gas turbine energy technologies are one of the most important components of the modern and advanced energy industry. One of the reasons for this situation is the role of gas in the fuel and energy balance of the largest economies. Thus, the share of gas in Russia's fuel balance is approximately 50%, and for other major players — the USA and the EU — 40% and 20%, respectively. Installing a combined-cycle power plant in the energy sector can already be a profitable option and generate income. The total share of gas fuel use in the global electric power industry is increasing linearly. It may reach up to 30% of all renewable resource options by 2070, compared with the indicator of 1965, which was only 15%. Gas turbine energy technologies can potentially play an important role in the next decade due to the transition from single—purpose single-fuel power plants to multi-purpose multi-fuel energy chemical complexes, in which high-quality equipment will be a key element - high-power gas power turbines with high inlet temperatures [1].

The concentration of a significant amount of natural fuel resources in Russia — about 20% of natural gas and coal, 13% of oil from the total world reserves — has made it possible to ensure a stable level of fuel supply to the population and industrial needs of the country. At the same time, a significant number of thermal power plants already in operation continue to effectively function mainly by natural gas and reliably meet growing demand using existing basic equipment [2].

Currently, gas turbine installations are not limited to use only in the field of electric power,

*CONTACT: B. Deepanraj, e mail: babudeepan@gmail.com

either as a separate key element of a modern power plant or as a part of a combined cycle plant, but also used in shipbuilding, the aviation industry, long-range gas supply, heating installations and the railway industry [3].

Therefore, the issue of ensuring the reliability and uninterrupted operation of equipment during the operational phase remains sensitive. This problem is becoming especially critical in modern conditions, which are associated with the following factors [4]:

- Rapid growth in the complexity of multi-object systems;
- Increasing complexity of the performed functions;
- Increased risks associated with equipment downtime;
- Increased strict requirements for meeting occupational safety requirements;
- Increased strict requirements for environmental protection.

An analysis of the operating experience of various types of units shows that maintenance and repair account for 15-12% of the calendar time (3-4% of which is occupied by unscheduled repairs), and their implementation is associated with high material costs. Maintenance and repair costs are one of the most important operational indicators of any technical system. Minimizing them in cases where the system is maintainable is practically impossible without effective monitoring and diagnosis of its condition. It also helps to reduce unexpected equipment failures, resulting in increased reliability of the equipment in operation [5, 6].

It is important to control the parameters of various components of a gas turbine installation (turbine, compressor, combustion chamber and bearings) that directly affect the whole operation of the engine. As well as their particular elements affect the operation of other gas turbine components, because a change in the efficiency of one of the components will change the efficiency of other components and the efficiency of the whole engine [7].

A classic gas turbine engine (GT) consists of: an air compressor, a combustion chamber and a gas turbine, as well as auxiliary systems that ensure its operation [8].

Through the air compressor, air from the atmosphere enters the combustion chamber at high pressure, along with the primary fuel. This interaction initiates the combustion process. The resulting mixture of gases (or combustion products) enters the GT turbine as a stream of incandescent gases, which causes its shaft to rotate due to the interaction of the turbine blades and the gas flow. The generated energy from rotation comes from the turbine shaft to the compressor and an electric generator, from the terminals of which, then, electrical energy is supplied [8].

Gas turbines operate according to the Brighton cycle, in which air becomes compressed and then fuel is added to it for subsequent ignition in order to generate energy. Important aspects in the operation of a gas turbine are the pressure ratio at which the inlet pressure is compared with the outlet pressure; ambient temperature, which affects the combustion of fuel in the turbine; turbine inlet temperature; and the intercooler, which is used to cool compressed air in the intercooler recovery system [9].

There are several methodological approaches to investigating equipment failures, including the FTA (Fault Tree Analysis) [10] and FMEA (Failure Mode and Effects Analysis) [11] methods. Modern control systems using industrial tools such as SCADA systems (Supervisory Control and Data Acquisition) and PLC (Programmable Logic Controller) allow obtaining information from numerous sensors measuring physical parameters, which allows for comprehensive diagnostics. Machine learning classification models are also used in many applications in industries [12]. Moreover, it becomes possible to identify non-obvious patterns and detect anomalies in order to make a complete and more accurate forecast based on all available aggregate information using machine learning and deep learning [13]. Thus, predictive accuracy is improved.

Fahmi et al. [13] proposed a combined architecture based on Temporal Convolutional Neural Networks (TCN) and an Autoencoder. The proposed solution made it possible to solve the problem of detecting anomalies in the data of a time series of indicators of sensors of a gas turbine installation.

Hanachi et al. [14] provide a comprehensive review of methods for monitoring, diagnosing, and predicting the technical condition of gas turbine engines, with a focus on performance data. One highlighted approach is the use of individual models, which are developed based on physical laws and operational indicators of the turbine. As an example of data-driven modelling, the authors discuss artificial neural networks in detail while also filtering a range of machine learning and deep learning methods. In addition, they mention fuzzy logic, Adaptive Neuro-Fuzzy Inference Systems (ANFIS), and Support Vector Machines (SVM) as commonly applied diagnostic and forecasting techniques. For forecasting, the study further cites models based on the Monte Carlo method, Markov chain theory, autoregressive models, and the Elman Artificial Neural Network (Elman ANN).

Saj et al. [4] propose a neural network architecture that combines a one-dimensional Convolutional Neural Network (Conv1D) with a recurrent Long Short-Term Memory (LSTM) network. In this design, the Conv1D layers serve to reduce data dimensionality and extract relevant features, thereby eliminating the need for manual feature engineering or specialized feature selection algorithms.

Amirkhani et al. [15] investigated the application of a series-parallel Nonlinear Autoregressive Exogenous (NARX) model for reliable fault detection in gas turbines. They evaluated four variants of the NARX model—MLP-NARX, RBF-NARX, GRNN-NARX, and ANFIS-NARX. All the demonstrated solutions showed a low regression error after training, which indicates the successful application of the proposed method.

Nashed et al. [16] investigated acoustic emission analysis of gas turbine engines by converting acoustic signals into images and classifying them using a ResNet-50 convolutional neural network. The data were preprocessed with a Wavelet transform and various filters, enabling effective image-based classification of acoustic emission signals.

This paper investigates the practical application of various artificial intelligence and machine learning algorithms for predicting gas turbine equipment failures. To address the data imbalance issue in the original data, data balancing methods like Tomek Links and the Synthetic Minority Over-Sampling Technique were employed. The classification methods considered include logistic regression, categorical boosting, random under-sampling boosting, echo-state network, ResNet, and a conditional variational autoencoder with a fuzzy logic layer.

2. DESCRIPTION OF DATASET

The dataset under study contains 10,000 non-zero rows of real-type data, representing sensor readings collected in various engine operating modes; 11 non-target features that directly reflect sensor readings; and 1 target feature showing the engine condition. This dataset is designed to detect and classify engine malfunctions [17]. Table 1 shows the characteristic non-target features of the original dataset [17].

Table 1. The characteristic non-target features

№	Parameter	Unit	Range
1	Peak vibration amplitude measured in the engine	mm/s²	0.1-10.0
2	Root Mean Square (RMS) of engine vibration	mm/s²	0.05-5.0
3	Frequency of engine vibration	Hz	20-2000
4	Temperature of the engine surface	°C	30-150
5	Temperature of the exhaust gas	°C	200-600
6	Acoustic noise level generated by the engine	dB	60-120
7	Acoustic signal frequency of the engine	Hz	100-5000
8	Intake manifold pressure	kPa	90-120
9	Exhaust gas pressure	kPa	80-110
10	Energy of the signal in a specific frequency band (from Short-Time Fourier Transform)	Arbitrary Units	0.1-1.0
11	Average signal amplitude over specific time windows	Arbitrary Units	0.01-0.5

The target predicted indicator contains three possible engine conditions: normal (0), minor (1) and critical (2) faults. At the same time, the data is unbalanced and distributed as follows:

- 60% of all data belongs to class "0" (normal condition) and reflects the correct operation of the engine without any faults;
- 30% of all data belongs to class "1" (minor faults) and reflects engine operation with minor faults;
- 10% of all data belongs to class "2" (critical faults) and reflects engine operation under severe fault conditions.

At the same time, when considering the data in a dimension smaller than the original one, another dataset problem was identified — the overlap of classes. The visualization of data in a smaller dimension (in this case, in a two-dimensional space) obtained by the t-SNE dimension reduction method is shown in Fig. 1.

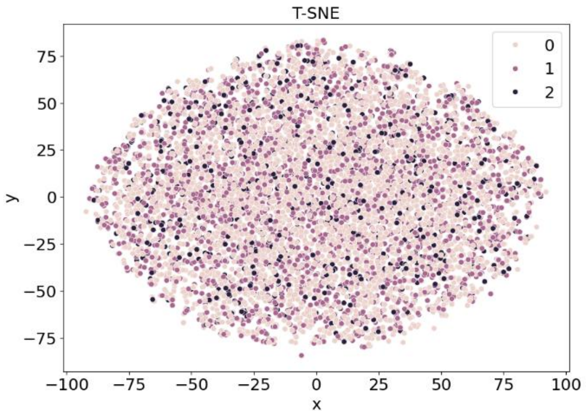

Fig. 1. Visualization of the source data in two-dimensional space

2.1 Data Preprocessing

For the experiments, Robust Scaling method was used for normalization. The original dataset was divided into training and test samples in a ratio of 70% to 30%, respectively. A training sample was used to train all the models, and a test sample was used to verify the prediction results of the final models. In addition, one-hot encoding of the target feature was performed for some machine-learning and deep-learning methods.

The prediction results of each of the solutions were evaluated using the following indicators: responsiveness, accuracy, F1 score, PR-AUC and ROC-AUC curves.

2.2 Data Preparation Methods

To reduce the impact of data imbalance problems and overlapping classes of the dataset used, the following sampling modification methods, in combination with the solutions discussed in this section.

Tomek Links (T-Links) is an incomplete sampling method developed by Tomek [18]. Initially, this method was considered for single-class classification and as an improvement of the nearest neighbour rule [19, 20]. A pair in Tomek's relations is such values of x and y (where x is an instance of class 0, y is an instance of class 1) for which there is no such value of z at which the inequality would be valid, as in (1), where d(x, y) is the distance between x and y:

$$d(x,y) < d(x,z) \; or \; d(x,y) < d(y,z). \qquad (1)$$

Then such a pair of Links is characterized by x and y belonging to noise or their location on the border of two classes. Therefore, the Tomek Links method can be used as a sampling reduction method, which removes instances of the majority class [19]. Elhassan et al. (19] and Swana et al. [21] also claim that the combination of this method with other methods of reducing and increasing the sample sometimes improves the prediction results and allows taking into account minor classes.

Over-sampling methods increase the number of minority instances by reproducing them, depending on the method used. The Synthetic Minority Over-Sampling Technique (SMOTE), based on the k-nearest neighbour method, is used to generate synthetic data of instances of the minority class based on pre-existing data of the same class and without duplication [21]. Unlike the classical sampling method, where examples of the minority class are either duplicated or reproduced randomly, the SMOTE algorithm creates new data by interpolating between several instances of the minority class located within a specific neighbourhood [22].

This method makes it possible to combat data overfitting due to the lack of duplication of generated examples, and improves the prediction of various models on unbalanced data [21].

3. MATERIALS AND METHODS

3.1 Stages of the Classification Process

The classification process is divided into the following stages: data preprocessing, data balancing (if necessary for a specific method), model training and tuning, and model evaluation. Fig. 2 demonstrates the diagram that represents the whole classification process in stages.

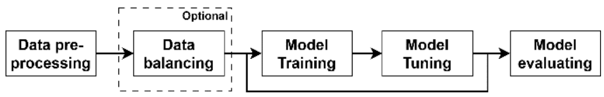

Fig. 2. The diagram that represents the whole classification process in stages

3.2 The Initial Proposed Solution

The initial dataset also included a predefined solution to the classification problem, as described in Kaggle [23], where the naive Bayesian classifier (GaussianNB) was employed as the model. Following the training process, the classifier yielded the following performance metrics: Recall = 0.333, Precision = 0.198, F1-score = 0.249, PR-AUC = 0.4035, and ROC-AUC = 0.500. The corresponding confusion matrix for this model is presented in Fig. 3.

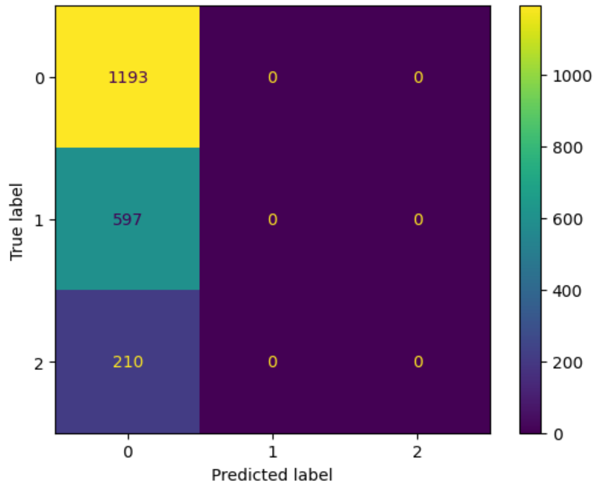

Fig. 3. Confusion matrix for GaussianNB model prediction results

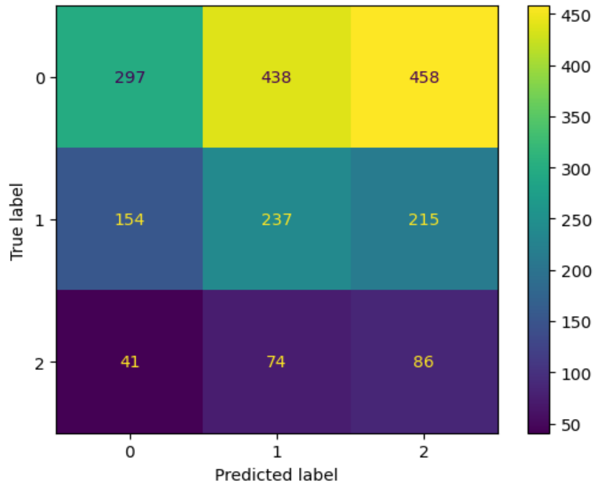

Fig. 4. Confusion matrix for Tomek Links + SMOTE + logistic regression model prediction results

Based on the results, it was concluded that the proposed solution demonstrates a complete disregard for minority classes and shows quality ratings of no more than 40%.

3.3 Logistic Regression

One of the most popular classification methods in machine learning is logistic regression. This method was used to compare it with the original one proposed because it also allows us to take into account class weights when predicting a label.

Before training the logistic regression model, the Tomek Links method of reducing examples of the majority class was applied, and the increase in instances of minority classes using the SMOTE sampling method was subsequently performed on the prepared data.

When predicting the trained model, the following metric values were obtained: Recall = 0.356, Precision = 0.345, F1-score = 0.293, PR-AUC = 0.405, ROC-AUC = 0.503. Fig. 4 demonstrates the confusion matrix for the prediction results of this model in the test sample. Based on the results, it was concluded that the proposed solution demonstrates a complete disregard for minority classes and shows quality ratings of no more than 40%.

3.4 Implementation of Gradient Boosting Categorical Boosting (CatBoost)

Gradient boosting is one of the most powerful machine learning methods that enables achieving high results in solving various practical tasks. The process of constructing an ensemble predictor provides its operation by performing gradient descent in a functional space. Categorical Boosting (CatBoost) is an implementation of gradient boosting that uses binary decision trees as basic predictors and copes well with categorical features [24]. The algorithm differs from other implementations of gradient boosting in the following aspects [25]:

- Automatic processing of categorical features as numerical characteristics;
- Using a combination of categorical features, taking advantage of the relationships between objects;
- Using models of perfectly symmetrical trees can reduce overfitting and increase the generalizing accuracy of the algorithm.

According to Prokhorenkova et al. [24] and Luo et al. [25], this algorithm shows better results compared to its analogues XGBoost and LightGBM.

The technical implementation of CatBoost also allows for the flexible customization of the model's training. It allows selecting a specific loss function, a metric for predicting model quality, various settings for the used predictor of the tree (such as its depth), adding appropriate regularization, as well as several advanced options for adjusting weights to improve the susceptibility of the model to minority classes. Before training the CatBoost model, the Tomek Links method of reducing examples of the majority class was applied, and the systematic increase in instances of minority classes using the SMOTE sampling method was subsequently performed on the prepared data.

Table 2 shows the results of experiments on training the CatBoost model using various combinations of learning rate and depth parameters.

Table 2. Results for different parameters combinations

Recall Precision F1-score		Learning rate				
		0.1	**0.01**	**0.001**	**0.0001**	**0.00001**
Depth	**4**	0.337 0.338 0.331	0.327 0.334 0.312	0.332 0.336 0.307	0.343 0.347 0.310	0.346 0.347 0.310
	5	0.336 0.336 0.334	0.328 0.334 0.317	0.328 0.331 0.307	0.346 0.346 0.313	0.347 0.348 0.313
	6	0.354 0.353 0.353	0.338 0.340 0.332	0.334 0.335 0.315	**0.355** **0.351** **0.322**	0.350 0.349 0.318
	7	0.337 0.338 0.338	0.339 0.339 0.337	0.321 0.324 0.309	0.342 0.342 0.317	0.344 0.344 0.318
	8	0.323 0.323 0.323	0.335 0.335 0.335	0.325 0.328 0.317	0.341 0.340 0.322	0.337 0.339 0.319
	9	0.320 0.319 0.320	0.324 0.323 0.323	0.330 0.329 0.325	0.340 0.340 0.327	0.341 0.340 0.326
	10	0.321 0.321 0.321	0.322 0.321 0.321	0.332 0.330 0.329	0.332 0.334 0.325	0.337 0.339 0.329

In this article, the CatBoost model was trained on a prepared dataset with the following best parameters shown in Table 3.

Table 3. Used CatBoost hyperparameters

№	Parameter	Value
1	Learning rate	0.0001
2	Loss function	MultiClassOneVsAll
3	Evaluation metric	HingeLoss
4	Class weight method	Balanced
5	L2 regularization	0.1
6	Max tree depth	6
7	Number of iterations	5000

The prediction results of the trained model were evaluated using the previously mentioned metrics, which showed the following estimates: Recall = 0.355, Precision = 0.351, F1-score = 0.322, PR-AUC = 0.407, ROC-AUC = 0.506. Fig. 5 demonstrates the confusion matrix for the prediction results of this model in the test sample.

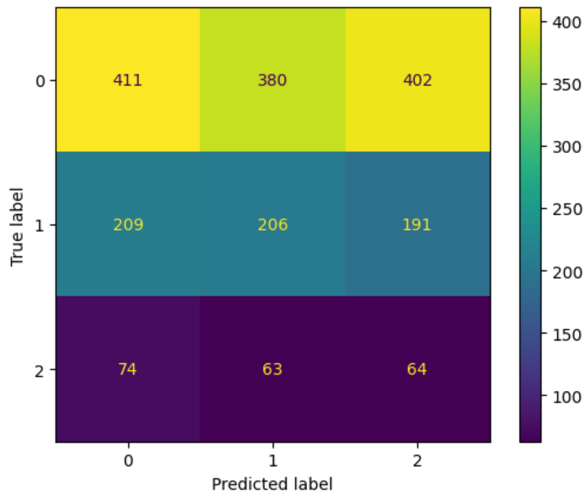

Fig. 5. Confusion matrix for Tomek Links + SMOTE + CatBoost model prediction results

Due to the flexible configuration of the model, as well as the built-in processing of categorical features and the consideration of class weights, it was possible to achieve results that exceeded the initial proposed solution for the data set under study. At the same time, the confusion matrix illustrates the continued imbalance in data forecasting, still in favour of the majority class.

3.5 Random Under-Sampling Method + Gradient Boosting (RUSBoost)

The solution underlying this type of gradient boosting, the Random Under-Sampling method, is one of the most common data sampling methods due to its simplicity: deleting examples of the majority class occurs randomly until the desired class distribution is achieved. The presented variant of gradient boosting, RUSBoost, is based on a similar algorithm, SMOTEBoost, which, in turn, improves the variant of gradient boosting, AdaBoost. RUSBoost differs from its predecessor in a simpler algorithm, which helps to achieve a faster learning rate of the model and higher performance [26].

To eliminate the difference between the methods, the data was processed using a combination of Tomek Links + SMOTE sampling methods. As noted earlier, AdaBoost was used as the basis for boosting, while the classical decision tree for the classification problem was used as an internal evaluation model.

A Cross-Validation grid search (GridSearchCV) was applied to select the hyperparameters of the model. Intervals were set for selecting accelerator and evaluator hyperparameters, from which the best parameters were selected. Detailed information about the final parameters is given in Table 4.

Table 4. Used RUSBoost hyperparameters

No.	Parameter	Range	Best
1	Max tree depth	(5, 10, 15)	15
2	Min samples split	(3, 5, 10)	3
3	Min leaf number	(10, 15, 20)	10
4	Number of estimators	(50, 100, 150, 200)	200
5	Learning rate	(0.1, 0.25, 0.5)	0.5

When predicting the trained model, the following metric values were obtained: Recall = 0.355, Precision = 0.359, F1-score = 0.354, PR-AUC = 0.412, ROC-AUC = 0.512. Fig. 6 demonstrates the confusion matrix for the prediction results of this model in the test sample.

Thanks to a combination of methods for changing the initial sample, namely the Tomek Links + SMOTE sequence + the Under-Sampling method built into the classifier, it was possible to achieve results that surpass the original proposed solution for the data set under study. At the same time, the confusion matrix illustrates the continued imbalance in data forecasting still in favour of the majority class. However, based on the same error matrix, we concluded that, despite better metrics than the CatBoost solution, this solution does not take into account minority classes.

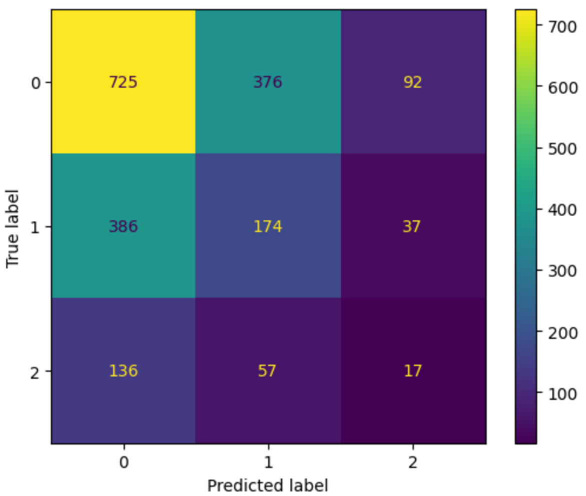

Fig. 6. Confusion matrix for Tomek Links + SMOTE + RUSBoost model prediction results

3.6 Echo-State Network (ESN)

One of the considered existing solutions used recurrent neural networks (RNN), which excel at predicting values or classes based on a specific time sequence, but their training is complex and computationally and time-consuming. As an alternative approach to learning, there is a way to use recurrent neural networks with Echo-State Networks (ESN).

An echo state network is a fast and efficient recurrent neural network that consists of an input layer, a recurrent layer, called in ESN terminology a "reservoir" that contains a large number of sparsely connected neurons, and an output layer. The weights of the input layer and reservoir connections are fixed after initialization, while the output weights, on the contrary, can be trained by solving the linear regression problem [27].

The effectiveness of this approach stems from training only the weights of the output data, without altering the weights of the input layer and reservoir. This approach allows for faster training times and reduced computational costs. At the same time, studies of reservoir computing have shown that in some cases, this approach does not lose accuracy in comparison with classical RNNs [27].

Just as for previous solutions, the Tomek Links method of reducing examples of the majority class was applied, and the increase in instances of minority classes using the SMOTE sampling method was subsequently performed on the prepared data.

In this paper, the ESN model was trained on a prepared dataset with the following parameters: number of neurons = 1000; leak rate = 1.0; spectral radius = 1.3; ridge = 1e-1.

When predicting the trained model, the following metric values were obtained: Recall = 0.3795, Precision = 0.3795, F1-score = 0.3795, PR-AUC = 0.360, ROC-AUC = 0.535. Fig. 7 demonstrates the confusion matrix for the prediction results of this model in the test sample.

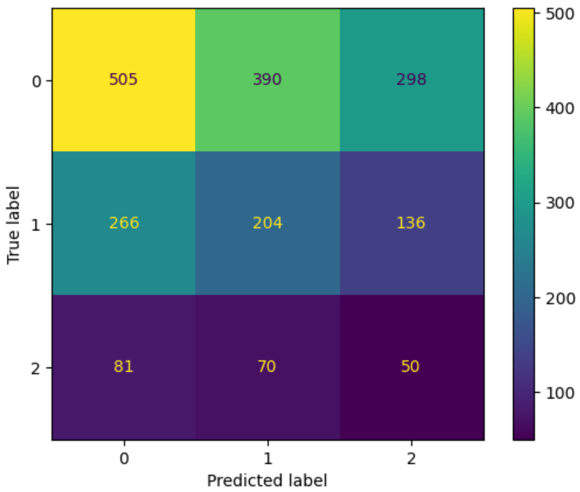

Fig. 7. Confusion matrix for Tomek Links + SMOTE + ESN model prediction results

Thanks to a combination of methods for changing the initial sample, namely the Tomek Links + SMOTE sequence + the random sample reduction method built into the classifier, it was possible to achieve results that surpass the original proposed solution for the data set under study. There is a noticeable improvement in the numerical values of all the considered metrics relative to the previously presented solutions. However, at the same time, the confusion matrix illustrates the continued imbalance in data forecasting, still in favour of the majority class.

3.7 Architecture of the ResNet Convolutional Network

The architecture of the ResNet convolutional neural network is well known in the field of image processing. Meanwhile, Gorishniy et al. [28] reported that such an architecture can be an effective basic model for solving the problem of processing tabular data.

The main advantage of this architecture is the use of Residual Blocks, which avoid the problem of a decaying gradient inherent in many deep neural networks [29]. The residual block enables enhanced feature extraction, improved performance, and the transfer of information through deep neural network connections, thereby facilitating the construction of deeper networks.

Unlike previous solutions, this method does not employ methods for reducing or increasing the sample, as it works by representing data in a hidden space. Even under these conditions, the results were comparable to the previous solutions considered.

In this paper, the ResNet model was trained on a prepared dataset with the following parameters: optimizer — Adam; learning rate = 3e-4; batch size = 64; epochs = 200.

When predicting the trained model, the following metric values were obtained: Recall = 0.343, Precision = 0.361, F1-score = 0.279, PR-AUC = 0.438, ROC-AUC = 0.619. Fig. 8 demonstrates the confusion matrix for the prediction results of this model in the test sample.

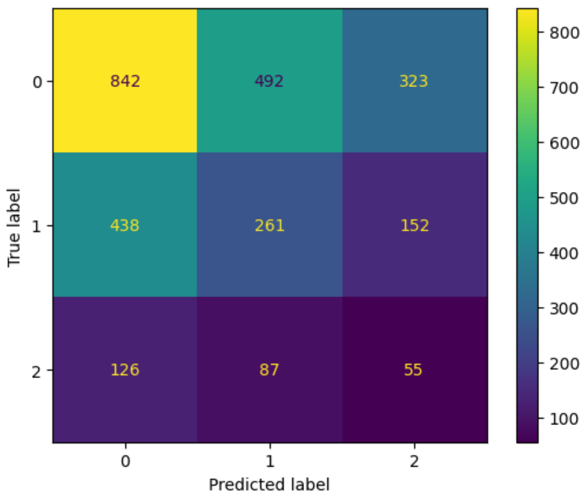

Fig. 8. Confusion matrix for ResNet-34 model prediction results

3.8 Conditional Variational Autoencoder (CVAE) + Fuzzy Logic Classifier Layer (FuzzyLayer)

The solution combination presented in [30] includes the use of a Variational Auto Encoder (VAE) and a Fuzzy Logic Layer (Fuzzy Layer). Variational Autoencoders are one of the most popular generative networks for studying data representation and use. Unlike traditional Autoencoders, they represent source data in a small latent space, which makes it possible to identify additional hidden characteristics of multidimensional data. Fuzzy logic systems, in turn, are excellent tools for data interpretability due to their structure based on specific rules and assumptions [30].

Bölat and Kumbasar [30] proposed an integrated framework that combines deep learning with fuzzy logic by first employing a Variational Autoencoder (VAE) to represent multidimensional data in a latent space for extracting semantic features. The latent space is then clustered using a fuzzy logic system to generate fuzzy sets, which serve as the foundation for building a fuzzy classifier. This classifier is subsequently trained using standard deep learning techniques to enhance classification performance.

The proposed approach was evaluated on the MNIST dataset of handwritten digits, where it achieved satisfactory results, demonstrating the effectiveness of combining VAEs, fuzzy clustering, and deep learning-based classification within a unified structure.

For this work, the solution was adapted to process tabular data. In the solution structure, the Variational Autoencoder was replaced with a Conditional Variational Autoencoder, since it additionally accepts a class label as input.

In this paper, the CVAE + FuzzyLayer model was trained on a prepared dataset with the following parameters: optimizer — Adam; learning rate = 3e-4; batch size = 150; epochs = 350.

When predicting the trained model, the following metric values were obtained: Recall = 0.355, Precision = 0.355, F1-score = 0.355, PR-AUC = 0.421, ROC-AUC = 0.6025. Fig. 9 demonstrates the confusion matrix for the prediction results of this model in the test sample.

Unlike previous solutions, the methods of reducing and increasing the sample were not used, since the process of presenting data in a latent space implements this process during training. Even under these conditions, the results were comparable to the previous solutions considered.

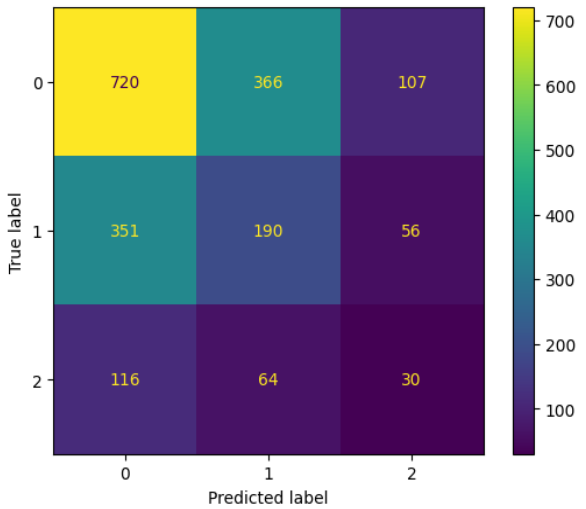

Fig. 9. Confusion matrix for CVAE + FuzzyLayer model prediction results

4. RESULTS AND DISCUSSION

Table 5 presents the results of experiments conducted on the original dataset using various models, along with the initial solution [23] provided alongside the dataset.

It was decided not to consider the ResNet neural network model, since its intended use is image processing, which may cause additional difficulties when implemented in real production. At the same time, its training takes the most computing time and resources.

Based on the presented table, it was concluded that all solutions show approximately the same results, not exceeding an average of 50%. Among all the solutions, ESN and CVAE+FuzzyLayer methods proved to be the best strategies, although the last of these solutions, as mentioned earlier, does not use sampling modification methods to eliminate data imbalances and produce results comparable to solutions that, in turn, use these methods. However, all solutions demonstrate a serious bias in the prediction results in favour of the majority class, which is clearly visible in the error matrices shown in Figure 3-9. The best solution based on this metric is the CatBoost model using the Tomek Links and SMOTE sampling methods. This highlights the importance of integrating data balancing techniques with robust classifiers to achieve more reliable and unbiased predictions.

Table 5. Results for different strategies

Strategies	Recall	Precision	F1-score	PR-AUC	ROC-AUC
GaussianNB	0.333	0.198	0.249	0.404	0.5
Logistic regression	0.356	0.345	0.293	0.405	0.503
CatBoost	0.355	0.351	0.322	0.407	0.506
RUSBoost	0.355	0.359	0.354	0.412	0.512
ESN	**0.380**	**0.380**	**0.380**	0.360	0.535
ResNet34	0.343	0.361	0.279	0.438	0.619
CVAE+Fuzzy Layer	0.355	0.356	0.355	**0.421**	**0.603**

In addition, the Friedman test and subsequent studies using the Nemenyi test were used to verify statistical significance, as described by Demšar [31]. To do this, 15 experiments on the models under consideration using cross-validation were additionally performed. Recall was used as a metric for evaluating model results. Table 6 demonstrates the results of 15 new experiments on the original dataset using various models. In all required calculations, the value 0.05 was used as the threshold p-value.

According to the Friedman test, the null hypothesis was rejected, indicating that all algorithms are not equivalent. Next, a post-hoc test was performed using the Nemenyi test. According to this test, the critical difference between the classifiers should be 1.575. Two groups of algorithms were identified: Logistic regression and CatBoost, and RUSBoost and CVAE + FuzzyLayer model – have a clear difference between each other. At the same time, it is not entirely clear which group the ESN algorithm consistently belongs to.

Obviously, such solutions cannot be used in real production with their current configuration, and therefore, a new challenge arises to develop new methods capable of solving the task of predicting the failure of gas turbine equipment with the problems inherent in the dataset under study and many other datasets based on real indications of industrial operation of a gas turbine. This highlights the urgent need for robust, scalable, and generalizable predictive models that can be reliably deployed in practical industrial environments.

Table 6. Results for 15 new experiments using cross-validation

No.	Logistic regression	CatBoost	RUSBoost	ESN	CVAE + Fuzzy
1	0.305 (5)	0.3485 (4)	0.465 (2)	0.394 (3)	0.592 (1)
2	0.3045 (4)	0.3545 (3)	0.4585 (1)	0.3925 (2)	0.1 (5)
3	0.3255 (5)	0.364 (4)	0.4595 (1)	0.4025 (3)	0.458 (2)
4	0.3255 (5)	0.358 (4)	0.459 (2)	0.38 (3)	0.596 (1)
5	0.3035 (4)	0.3535 (3)	0.4595 (1)	0.399 (2)	0.2275 (5)
6	0.303 (5)	0.3585 (4)	0.4605 (2)	0.4015 (3)	0.484 (1)
7	0.3115 (5)	0.367 (4)	0.458 (2)	0.3945 (3)	0.5105 (1)
8	0.3315 (4)	0.3725 (3)	0.4675 (1)	0.4005 (2)	0.1295 (5)
9	0.31 (5)	0.355 (4)	0.4695 (2)	0.376 (3)	0.573 (1)
10	0.2995 (5)	0.344 (4)	0.487 (2)	0.4005 (3)	0.538 (1)
11	0.3205 (5)	0.3565 (4)	0.4675 (1)	0.3985 (2)	0.375 (3)
12	0.3325 (5)	0.3545 (4)	0.456 (2)	0.3895 (3)	0.573 (1)
13	0.3205 (5)	0.38 (4)	0.459 (1)	0.4005 (3)	0.4125 (2)
14	0.295 (5)	0.3805 (4)	0.4605 (2)	0.385 (3)	0.544 (1)
15	0.3135 (5)	0.3425 (4)	0.4645 (2)	0.381 (3)	0.5385 (1)
Avg.	**4,8**	**3,8**	**1,6**	**2,73**	**2,07**

5. CONCLUSION

During experiments on the initial dataset using various models, it was concluded that the simultaneous problems of data imbalance and class overlap had a substantial effect on the final prediction values and, as a result, on the overall effectiveness of the models in solving the problem of predicting the failure of gas turbine equipment. The initial solution proposed, along with the dataset considered in the paper, does not address its specified problems and completely overlooks the two minor classes, which, despite a generally satisfactory accuracy indicator, suggest poor model efficiency. The remaining solutions, on the contrary, show approximately similar results for other metrics, but they take into account all available classes in their forecasts. However, their efficiency and accuracy are still not at a high enough level, indicating the need for a more comprehensive and in-depth approach to solving the problem using the dataset in question. It is necessary to develop new methods and algorithms for the successful prediction of gas turbine engine failure based on real data, with possible problems of data imbalance and class overlap.

Further research will focus on advanced data preprocessing techniques to address issues of separability and imbalance, as well as exploring hybrid approaches that combine different machine learning methods for predicting gas turbine failures. Additionally, evaluating the computational cost and model complexity, which were not addressed in this study, will be a priority. Further research will be based on alternative data balancing methodologies to compare their effectiveness and identify the most suitable solution.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

[1] Y.K. Petrenya, Development of Gas Turbine Energy Technologies in Russia. *Herald of the Russian Academy of Sciences*, 89(2), 2019: 101-104. https://doi.org/10.1134/S1019331619020163

[2] A.A. Kudinov, Thermal power plants. Schematics and Equipment: Training Manual. *INFRA-M*, Moscow, 2015.

[3] V.G. Zlobin, A.A. Verholancev, Gas turbine installations. *Part 1. Thermal schemes. Thermodynamic cycles. Training Manual*, *St. Petersburg State University of Industrial Technologies and Design*, Saint Petersburg, 2020. (In Russian)

[4] W.C. Saj, M.V. Shcherbakov, A classification approach based on a combination of deep neural networks for predicting failures of complex multi-object systems. *Modeling, Optimization and Information Technology*, 8(2), 2020: 1-11. (In Russian) https://doi.org/10.26102/2310-6018/2020.29.2.037

[5] S. Dedyukhin, K.D. Andreev, Malfunction diagnostics of gas turbine units using vibrodiagnostics. *International Journal of Humanities and Natural Sciences*, 5(1), 2021: 16-25. (In Russian) https://doi.org/10.24412/2500-1000-2021-5-1-16-25

[6] A.D. Fentaye, A.T. Baheta, S.I. Gilani, K.G. Kyprianidis. A Review on Gas Turbine Gas-Path Diagnostics: State-of-the-Art Methods, Challenges and Opportunities. *Aerospace,* 6(7), 2019: 83. https://doi.org/10.3390/aerospace6070083

[7] R. Kurz, K. Brun, C. Meher-Homji, Gas Turbine Degradation. *43rd Turbomachinery & 30th Pump Users Symposia (Pump & Turbo 2014)*, 23-25 September 2014, Houston, USA, pp.1 -36. https://doi.org/10.21423/R15W5P

[8] V.V. Sakhin The design and operation of power plants. Book 2. Gas turbines. Heat exchangers: a textbook. *Ministry of Education and Science of the Russian Federation, Baltic State Technical University,* St. Petersburg, 2015.

[9] A.O. Onokwai, U.B. Akuru, D.A. Desai, Mathematical Modelling and Optimisation of Operating Parameters for Enhanced Energy Generation in Gas Turbine Power Plant with Intercooler. *Mathematics*, 13(1), 2025: 174. https://doi.org/10.3390/math13010174

[10] B. Novaković, Lj. Radovanović, D. Vidaković, L. Đorđević, B. Radišić, Evaluating Wind Turbine Power Plant Reliability Through Fault Tree Analysis. *Applied Engineering Letters*, 8(4), 2023: 175-182. https://doi.org/10.18485/aeletters.2023.8.4.5

[11] H. Ghasemian, Q. Zeeshan. Failure Mode and Effect Analysis (FMEA) of Aeronautical Gas Turbine using the Fuzzy Risk Priority Ranking (FRPR) Approach. *International Journal of Soft Computing and Engineering (IJSCE)*, 7(1), 2017: 81-92.

[12] A. Mishra, Machine Learning Classification Models for Detection of the Fracture Location in Dissimilar Friction Stir Welded Joint. *Applied Engineering Letters*, 5(3), 2020: 87-93. https://doi.org/10.18485/aeletters.2020.5.3.3

[13] A.-T.W.K. Fahmi, K. Reza Kashyzadeh, S. Ghorbani, Advancements in Gas Turbine Fault Detection: A Machine Learning Approach Based on the Temporal Convolutional Network– Autoencoder Model. *Applied Sciences*, 14(11), 2024: 4551. http://dx.doi.org/10.3390/app14114551

[14] H. Hanachi, C. Mechefske, J. Liu, A. Banerjee, and Y. Chen, Performance-Based Gas Turbine Health Monitoring, Diagnostics, and Prognostics: A Survey. *IEEE Transactions on Reliability*, 67(3), 2018: 1340-1363. http://doi.org/10.1109/TR.2018.2822702

[15] S. Amirkhani, A. Tootchi, A. Chaibakhsh, Fault detection and isolation of gas turbine using series-parallel NARX model. *ISA Transactions*, 120, 2022: 205-221. https://doi.org/10.1016/j.isatra.2021.03.019

[16] M.S. Nashed, J.R. Renno, M.S. Mohamed, R. Reuben, Gas turbine failure classification using acoustic emissions with wavelet analysis and deep learning. *Expert Systems with Applications*, 232, 2023: 120684. https://doi.org/10.1016/j.eswa.2023.120684

[17] Engine Fault Detection Data : Sensor data for engine condition classification and predictive maintenance. *Kaggle*, 2025. https://www.kaggle.com/datasets/ziya07/engine-fault-detection-data (Accessed: 15 April 2025)

[18] I. Tomek, An Experiment with the Edited Nearest-Neighbor Rule. *IEEE Transactions on Systems, Man, and Cybernetics*, 6(6), 1976: 448-452. https://doi.org/10.1109/TSMC.1976.4309523

[19] A.T. Elhassan, M. Aljourf, M. Shoukri, Classification of Imbalance Data using Tomek Link(T-Link) Combined with Random Under-sampling (RUS) as a Data Reduction Method. *Global Journal of Technology and Optimization*, S1, 2017: 111. https://doi.org/10.4172/2229-8711.S1111

[20] R.M. Pereira, Y.M.G. Costa, C.N. Silla Jr., MLTL: A multi-label approach for the Tomek Link under sampling algorithm. *Neurocomputing*, 383, 2020: 95-105. https://doi.org/10.1016/j.neucom.2019.11.076

[21] E.F. Swana, W. Doorsamy, P. Bokoro, Tomek Link and SMOTE Approaches for Machine Fault Classification with an Imbalanced Dataset. *Sensors*, 22(9), 2022: 3246. https://doi.org/10.3390/s22093246

[22] A. Fernández, S. García, J. Luengo, E. Bernadó-Mansilla, F. Herrera, Genetics-Based Machine Learning for Rule Induction: State of the Art, Taxonomy, and Comparative Study. *IEEE Transactions on Evolutionary Computation*, 14(6), 2010: 913–941. https://doi.org/10.1109/TEVC.2009.2039140

[23] Engine Fault Detection Prediction. *Kaggle*, 2025. https://www.kaggle.com/code/rohitkumar211987/engine-fault-detection-prediction Accessed: 15 April 2025)

[24] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, A. Gulin, CatBoost: unbiased boosting with categorical features. *arXiv*, 2019. https://doi.org/10.48550/arXiv.1706.09516

[25] M. Luo, Y. Wang, Y. Xie, L. Zhou, J. Qiao, S. Qiu, Y. Sun, Combination of Feature Selection and CatBoost for Prediction: The First Application to the Estimation of Aboveground Biomass. *Forests*, 12(2), 2021: 216. https://doi.org/10.3390/f12020216

[26] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, A. Napolitano, RUSBoost: A Hybrid Approach to Alleviating Class Imbalance. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 40(1), 2010: 185-197. https://doi.org/10.1109/TSMCA.2009.2029559

[27] C. Sun, M. Song, S. Hong, H. Li, A Review of Designs and Applications of Echo State Networks. *arXiv*, 2020. https://doi.org/10.48550/arXiv.2012.02974

[28] Y. Gorishniy, I. Rubachev, V. Khrulkov, A. Babenko, Revisiting Deep Learning Models for Tabular Data. *Proceedings of the 35th International Conference on Neural Information Processing Systems,* New York, USA, 2021: 1447.

[29] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, Las Vegas, USA, pp.770-778. https://doi.org/10.1109/CVPR.2016.90

[30] K. Bölat, T. Kumbasar, Interpreting Variational Autoencoders with Fuzzy Logic: A step towards interpretable deep learning based fuzzy classifiers. *2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2020)*, 19-24 July 2020, Glasgow, UK. pp.1-7. https://doi.org/10.1109/FUZZ48607.2020.9177631

[31] J. Demšar. Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, 7, 2006: 1-30.