CARAVEL

Publish open access at Caravel Press

Research and Reviews in Sustainability

Journal homepage: sustainability-journal.com

Technical article

Predictive Modeling of Energy and Exergy Effects from Injector Placement in HCCI Engines Running on Diethyl Ether and Biogas Using Machine Learning Techniques

Aditya Sai Samavedam [1], Prasshanth CV [1], Manavalla Sreekanth [1], Tamilselvan P [1] and Feroskhan M [1]*

[1] *School of Mechanical Engineering, Vellore Institute of Technology, Chennai, India*

A B S T R A C T

The global emphasis on sustainable energy has highlighted biogas as a viable alternative fuel for internal combustion engines. Among advanced engine technologies, Homogeneous Charge Compression Ignition (HCCI) engines stand out due to their notable efficiency and low emissions, making them well-suited for clean energy applications. A key aspect in maximizing the benefits of biogas in HCCI engines, especially when paired with diethyl ether (DEE) as a pilot fuel, involves determining the optimal injector location. This study employed machine learning techniques to identify the most effective injector position by analyzing variables such as engine load, biogas flow rate, methane content, and intake air temperature. Three injector configurations were tested: one at the intake port and two further upstream at 6 cm (Manifold 1) and 10 cm (Manifold 2) from the intake. To solve this complex optimization problem, five advanced machine learning models SVM, Random Forest, AdaBoost, CATBoost, and XGBoost were utilized to predict crucial performance metrics, including brake-specific energy use, exergy losses, thermal efficiency, fuel consumption, and exergy-based efficiency. XGBoost emerged as the most accurate model, consistently delivering the highest R^2 values. The findings underscore the potential of machine learning to enhance injector placement strategies, thereby improving the energy efficiency and overall performance of HCCI engines operating on biogas and DEE.

A R T I C L E I N F O

Article History:
Received: 21 November 2025
Revised: 17 December 2025
Accepted: 10 January 2026

Keywords:
Biogas
Diethyl Ether (DEE)
Homogenous Charge Compression Ignition Engine (HCCI)
Injector Location
Machine Learning

Article Citation:
Samavedam, A. S., CV, P., Sreekanth, M., P, T., & M, F. (2026). Predictive Modeling of Energy and Exergy Effects from Injector Placement in HCCI Engines Running on Diethyl Ether and Biogas Using Machine Learning Techniques. Research and Reviews in Sustainability, 2(1), 62–84. https://doi.org/10.65582/rrs.2026.006

1. INTRODUCTION

Growing global interest in sustainable energy has increased attention on renewable fuels like biogas, recognized for being a carbon-neutral choice for powering internal combustion engines. Produced through the anaerobic digestion of organic materials, biogas shows great promise, though conventional engines often struggle to balance efficiency with low emissions when using it. HCCI engines help address these challenges by integrating features from both spark-ignition and compression-ignition systems, leading to higher thermal

* Corresponding author. Email address: feroskhan.m@vit.ac.in (Feroskhan M)

efficiency and reduced emissions. A pivotal factor in optimizing biogas use in HCCI engines is determining the best injector position for pilot fuels such as diethyl ether (DEE), since it significantly influences metrics like brake thermal efficiency and exergy efficiency. While experimental research and CFD simulations have provided important insights, they may not effectively capture complex variable interactions. In this context, machine learning has become a valuable tool, offering the ability to uncover hidden patterns and accelerate the identification of optimal injector configurations for HCCI engines running on biogas and DEE.

Several researchers have explored machine learning (ML) techniques to predict and optimize the performance of various engine types, including HCCI engines. Hanuschkin et al. (2021) used ML to analyse in-cylinder flow fields and predict combustion engine performance, while Wong et al. (2013) applied advanced ML models to optimise biodiesel engines. Extending to CI engines, Ağbulut, Gürel and Sarıdemir (2021) employed ML for nanoparticle–diesel blends, and Wang et al. (2022) compared ML methods for gasoline Wankel engines.

In SI engines, Liu, Ulishney and Dumitrescu (2020, 2021) developed Random Forest-based ML models for natural gas engines, improving predictions of combustion metrics. ML has also been applied to gas turbines (Liu and Karimi, 2020) and extensively to HCCI engines. Çay et al. (2012) pioneered artificial neural network (ANN) models for alternative-fuelled engines, while Balki et al. (2018) and Çay (2013) expanded the use of ANNs in SI engines.

Furthermore, Oğuz, Saritas and Baydan (2010), Patel, Azad and Khan (2019), and Ghobadian et al. (2009) modelled diesel engine performance and emissions using biofuels and ANN approaches. For diverse biofuel blends, Sebayang et al. (2017, 2022) and Gökalp et al. (2010) employed extreme learning machines and ANNs to predict engine performance and exhaust emissions.

Researchers have diversified machine learning (ML) approaches beyond artificial neural networks (ANNs). Kullolli, Sakthivel and Ilangkumaran (2016) applied neural networks to investigate the effects of injection timing in compression ignition (CI) engines, while Correa Gonzalez et al. (2021) predicted ignition characteristics in gasoline blends. Lionus Leo et al. (2024) employed Random Forest models to analyse an HCCI–DI engine operating with biodiesel–nano additives.

Studies by Sahoo, Kumar and Srivastava (2022) and Liu and Liu (2022) demonstrated the adaptability of ML techniques to compressed natural gas (CNG) and ammonia-fuelled engines, respectively. Furthermore, Makalesi et al. (2022) and Zandie et al. (2023) highlighted the capability of ML models for multi-fuel exhaust emission prediction and multi-output engine performance modelling.

Advanced techniques such as long short-term memory (LSTM) networks were employed by Liu et al. (2022) for micro gas turbine engines fuelled with microalgae blends. Broader ML applications include the prediction of fuel consumption and maintenance costs (Katreddi, 2023) and hybrid physical–ML modelling approaches for diesel engine emission prediction (Mohammad et al., 2021). Finally, closely aligning with the present study, Kale and Krishnasamy (2024) utilised ML techniques for performance prediction and optimisation of HCCI engines operating on biofuel–gasoline blends, underscoring the growing role of ML in modelling complex engine systems.

Extending prior studies, this work employs advanced machine learning techniques to optimize injector placement and fuel–air preparation parameters for biogas and diethyl ether fuelled HCCI engines under varying engine loads. Three injector locations are examined, namely direct injection at the intake port and two upstream manifold positions located 6 cm (Manifold 1) and 10 cm (Manifold 2) from the intake. Five machine learning models, including SVM, Random Forest, AdaBoost, CatBoost, and XGBoost, are used to predict key performance and exergy-based indicators such as brake-specific energy consumption, brake thermal efficiency, fuel consumption, exergy losses, and overall exergy efficiency. Although prior studies have applied machine learning models to energy systems for performance prediction, most have focused on output estimation and have rarely incorporated exergy-based analysis or design-level parameters such as injector placement. This study addresses this gap by integrating machine learning with exergy evaluation to quantify the influence of injector location and operating conditions on combustion irreversibility and efficiency, thereby offering new insights for the optimization of renewable-fueled HCCI engines.

2. EXPERIMENTAL SETUP

The experimental framework, depicted in Figure 1, was established to investigate an HCCI engine running on simulated biogas alongside diethyl ether (DEE). This involved converting a Kirloskar 8 HP single-cylinder

compression ignition engine into a water-cooled HCCI configuration. The synthetic biogas was created by carefully mixing methane and carbon dioxide streams, which were then blended with intake air before entering the engine. DEE was introduced as a pilot fuel via injectors positioned both at the intake port and further upstream to explore different injection strategies. Power output was tracked using an eddy current dynamometer, while DEE consumption was precisely measured with a digital balance. Injection timing control was managed electronically. By systematically varying conditions such as engine load, biogas flow rate, methane concentration, intake air temperature, and DEE injection characteristics, a robust dataset was developed for training the machine learning models.

Figure 1. Experimental setup used in this study.

3. METHODOLOGY

This section outlines the comprehensive methodology employed to meticulously prepare the dataset, enabling the machine learning model to undergo effective training and subsequently deliver accurate predictions.

3.1. DATA ACQUISITION AND MACHINE LEARNING MODELLING

During the experimental phase, the engine operated on air and simulated biogas, with diethyl ether (DEE) used as a pilot fuel introduced through the intake manifold. DEE was supplied at three different points: directly at the intake port, 6 cm upstream (Manifold 1), and 10 cm further upstream from Manifold 1 (Manifold 2), as illustrated in Figure 2. To comprehensively assess engine performance, tests were carried out by systematically varying several intake parameters, including the injector location (port, Manifold 1, Manifold 2), methane concentration (60% CH_4–40% CO_2 and pure CH_4), and engine load levels of 5, 10, 15, and 20 Nm, while maintaining constant intake air and coolant temperatures of 35°C. The biogas flow rate was also varied across different operating conditions. Data collected included five input parameters namely injector position, methane fraction, engine load, intake temperature, and biogas flow rate and five output metrics namely brake specific energy consumption, brake thermal efficiency, brake specific fuel consumption, exergy loss, and exergy efficiency. In total, this experimental design resulted in 60 distinct engine test runs covering all combinations of the selected operating parameters. This dataset was used to train and validate multiple machine learning models to predict engine performance under diverse operating conditions and to identify the optimal injector location. These input parameters were selected as they directly affect mixture formation, combustion characteristics, and in cylinder thermodynamic behavior, which in turn govern exergy destruction and efficiency. Their inclusion enables the models to capture the influence of operating conditions and fuel composition on exergy based performance. The overall methodology adopted in this study, along with a correlation heatmap of all variables, is presented in Figures 3 and 4.

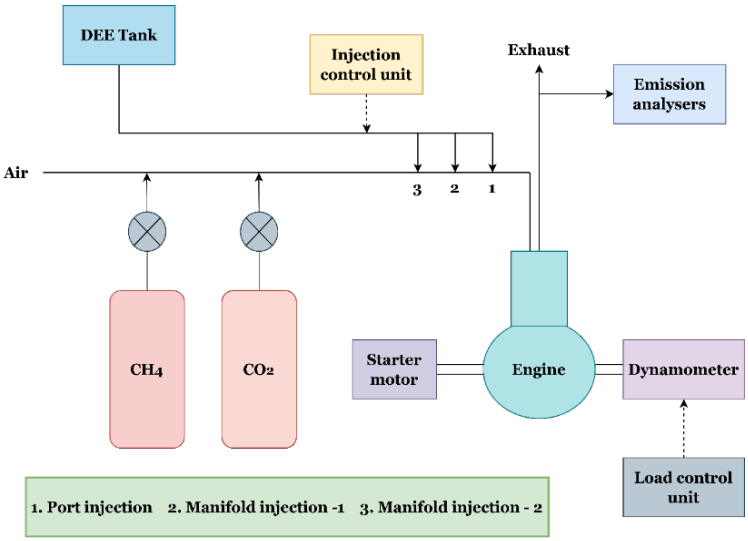

Figure 2. Layout of the experimental setup used in the study.

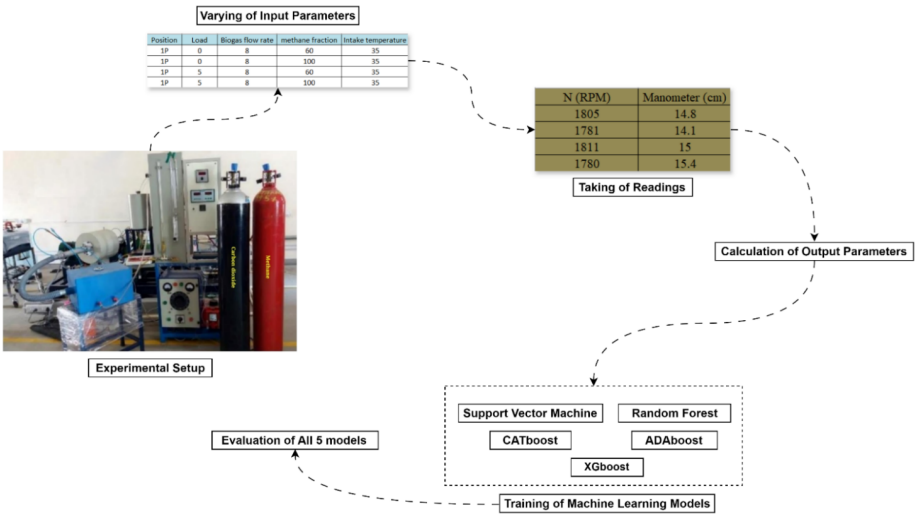

Figure 3. Workflow employed in the current study.

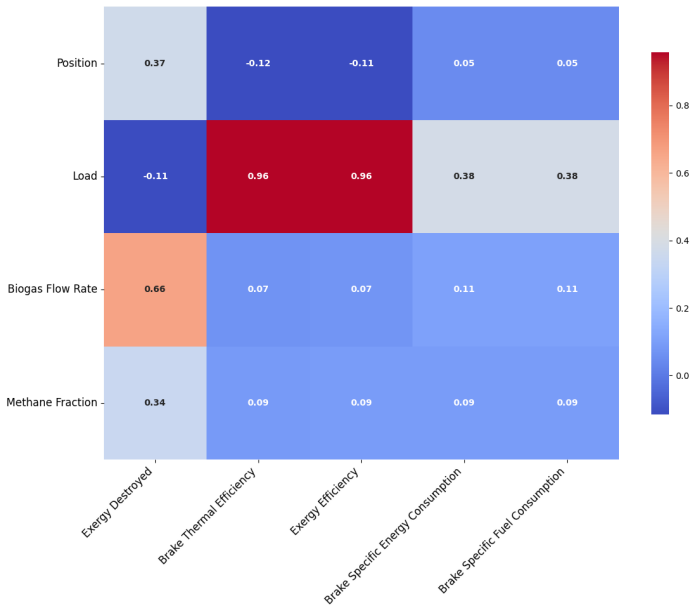

Figure 4. Correlation Heatmap for the input and output features.

3.2. PERFORMANCE ANALYSIS

This section outlines the comprehensive methodology employed to meticulously prepare the dataset, enabling the machine learning model to undergo effective training and subsequently deliver accurate predictions.

3.2.1. ENERGY ANALYSIS

In the comprehensive analysis, key parameters were meticulously evaluated to gain in-depth insights into the engine's performance capabilities. Through a thorough assessment, three critical energy parameters were identified as offering the most valuable opportunities for optimization and efficiency enhancement: brake power output, brake thermal efficiency, and Brake Specific Fuel Consumption.

1. Brake Power Output:

$$P_b = 2\pi NT \quad (1)$$

Where P_b is the brake power output, N is the engine speed in revolutions per second, and T is the torque measured by the dynamometer. This parameter quantifies the actual useful work output of the engine, serving as a direct measure of its overall performance. As such, it is an indispensable metric for evaluating the engine's capabilities

2. Brake Thermal Efficiency:

$$\eta_{th} = \frac{P_b}{m_f \cdot LHV} \quad (2)$$

Where η_{th} is the brake thermal efficiency, P_b is the brake power output, m_f is the mass flow rate of fuel, and LHV is the lower heating value of the fuel. This parameter characterizes the engine's ability to convert the chemical energy stored in the fuel into useful work. It provides invaluable insights into fuel utilization and energy conversion efficiency, making it a crucial aspect for identifying potential areas for improvement.

3. Brake Specific Fuel Consumption:

$$BSFC = \frac{\dot{m}_f}{P_b} \quad (3)$$

Where BSFC is the Brake Specific Fuel Consumption, $\dot{m}_f$ is the mass flow rate of fuel, and P_b is the brake power output of the engine. This parameter measures the fuel efficiency of the engine, indicating how much fuel is required to produce a certain amount of power. A lower BSFC value indicates a more efficient engine.

4. Brake Specific Energy Consumption (BSEC)

$$BSEC = \frac{m_f \cdot LHV}{P_b} \quad (4)$$

Where BSEC is the brake specific energy consumption, m_f is the mass flow rate of fuel, LHV is the lower heating value of the fuel, and P_b is the brake power output.

Brake Specific Energy Consumption (BSEC) quantifies the amount of fuel energy needed to generate one unit of brake power. It is determined based on the fuel mass flow rate, lower heating value (LHV), and brake power. While BSFC assesses fuel efficiency, BSEC offers a more complete picture by accounting for fuel energy content, making it useful for comparing different fuels. Together, these metrics provide valuable insight into engine efficiency and fuel utilization.

3.2.2. EXERGY ANALYSIS

To complement the energy analysis, exergy analysis was performed to evaluate engine irreversibilities and the quality of energy conversion. Two key exergy parameters were considered: exergy destruction and exergy efficiency, as they capture inefficiencies not reflected by brake power, brake specific fuel consumption, or brake thermal efficiency.

1. Exergy Destruction:

$$E_d = \dot{m}(h - T_0 s) - \dot{m}(h_0 - T_0 s_0) \quad (5)$$

Where $\dot{m}$ is the mass flow rate, h and s are the specific enthalpy and entropy of the working fluid, and T_0 is the reference environment temperature. Exergy destruction quantifies the loss of useful work due to irreversibilities such as combustion, heat transfer, and friction within the engine.

2. Exergy Efficiency:

$$\eta_{ex} = \frac{\text{Useful Exergy Output}}{\text{Total Exergy Input}} \quad (6)$$

This parameter evaluates the effectiveness of the engine in converting the available exergy of the fuel into

useful work. The exergy analysis is based on the assumptions of steady-state operation, negligible kinetic and potential energy effects, uniform reference environmental conditions, and idealized thermodynamic properties of the working fluid. Together, these exergy parameters provide a clearer understanding of internal losses and energy utilization, enabling identification of optimization potential across different operating conditions.

4. OBSERVED TRENDS

This section presents a detailed analysis of the key trends and patterns observed during the experimental investigation of HCCI engine performance using biogas and diethyl ether (DEE) as fuels, building upon observations reported in our previous study (Samavedam et al., 2024). All figures presented in this section are taken from (Samavedam et al., 2024).

4.1. TRENDS OF BRAKE SPECIFIC ENERGY CONSUMPTION

This subsection looks at how injector location, engine load, methane fraction, and biogas flow rate affect Brake Specific Energy Consumption (BSEC) in a biogas engine, based on detailed experiments. Tests were carried out with three injector positions: port injection, Manifold 1 (6 cm away), and Manifold 2 (10 cm away). As shown in Figure 5, BSEC drops steadily as engine load increases for all injector setups. Port injection consistently showed the lowest BSEC, indicating better fuel efficiency. Although Manifold 1 and 2 had higher BSEC, they followed the same trend. Interestingly, Manifold 2 often performed better than Manifold 1, even though it was placed farther from the port, which goes against the usual expectation that closer injection is always more efficient. Injector placement had the biggest effect at low loads, where port injection clearly outperformed the other two. At higher loads, the differences became smaller. Along with injector location and load, the study also examined how biogas flow rate and methane fraction influence BSEC. A higher biogas flow rate generally led to increased BSEC, but BSEC still went down with load, showing why it is important to fine-tune the flow rate. Surprisingly, increasing the methane fraction raised BSEC, as seen in Figures 6 and 7, which means the methane content also needs careful adjustment depending on engine load.

Overall, these results show that injector location, load, methane level, and flow rate all interact in complex ways to influence BSEC. Considering all these factors together is key to improving fuel economy and engine performance.

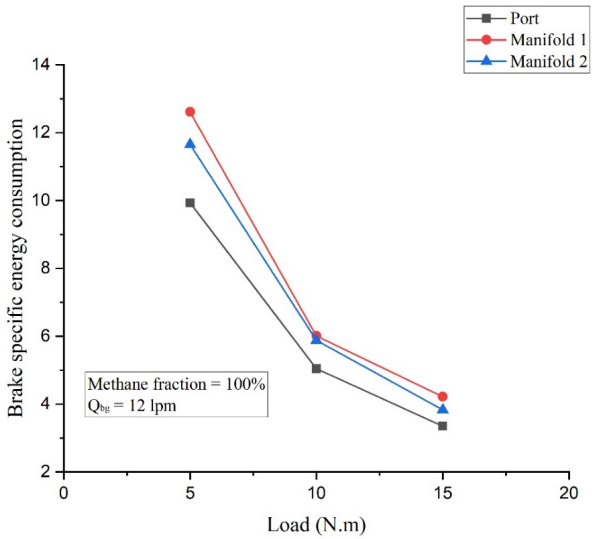

Figure 5. BSEC vs Load at various injector locations.

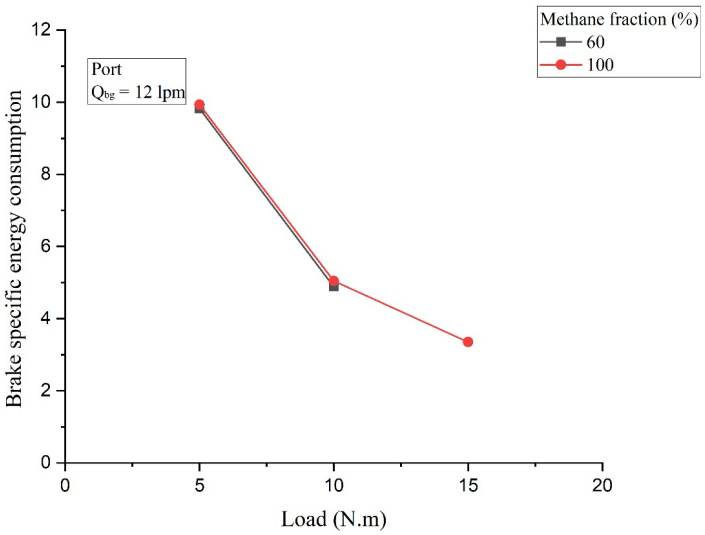

Figure 6. BSEC vs Load at various methane fractions.

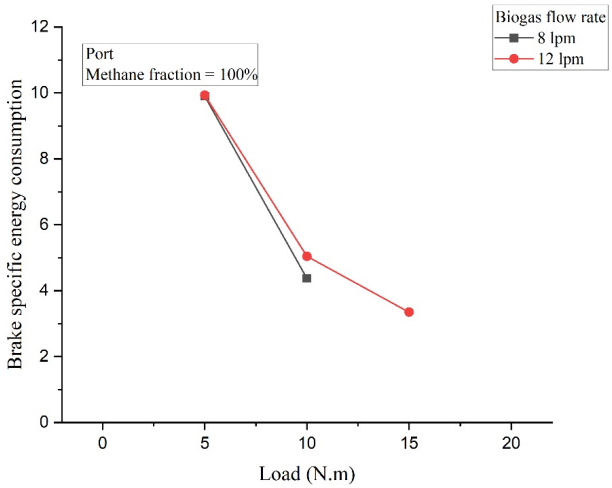

Figure 7. BSEC vs Load at various Biogas flow rates.

4.2. TRENDS OF BRAKE THERMAL EFFICIENCY

In the experimental study, it was observed that Brake Thermal Efficiency (BTE), also known as 1st law efficiency, serves as a crucial performance metric for internal combustion engines. It indicates how efficiently an engine transforms the chemical energy of the fuel into usable mechanical output, with higher values indicating better performance. The experiments involved three injector locations: Port injection, Manifold 1 (6 cm from the port), and Manifold 2 (10 cm from the port). Results indicated that BTE increases with engine load for all locations, a typical trend since engines generally operate more efficiently under higher loads. Port injection consistently demonstrated the highest efficiency, suggesting superior fuel-air mixing and combustion. Counterintuitively, Manifold 2 outperformed Manifold 1 despite being further from the port, indicating that the relationship between injector distance and efficiency is not linear as per Figure 8. The efficiency curves for all locations showed steeper increases at lower loads, with diminishing improvements at higher loads. This suggests that the greatest efficiency gains occur in the transition from low to medium loads. Furthermore, the experiment indicated that methane fraction and biogas flow rate significantly influence BTE. Higher methane fractions lowered efficiency when compared to lower methane fractions while increased biogas flow rates also led to lower BTE, especially at higher loads as per figures 9 and 10. The impact of these factors became more

pronounced as engine load increased.

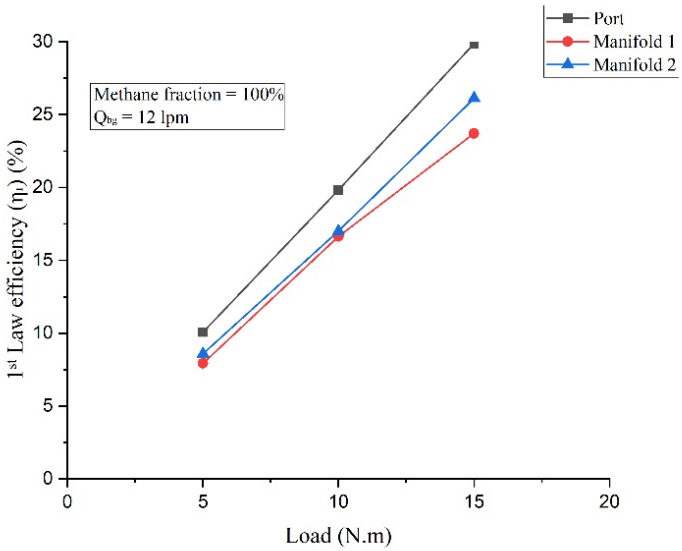

Figure 8. Brake thermal efficiency vs Load at different injector locations.

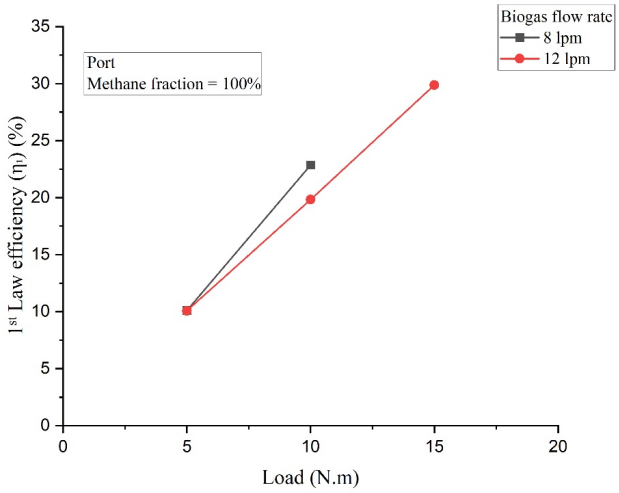

Figure 9. Brake thermal efficiency vs Load at different biogas flow rates.

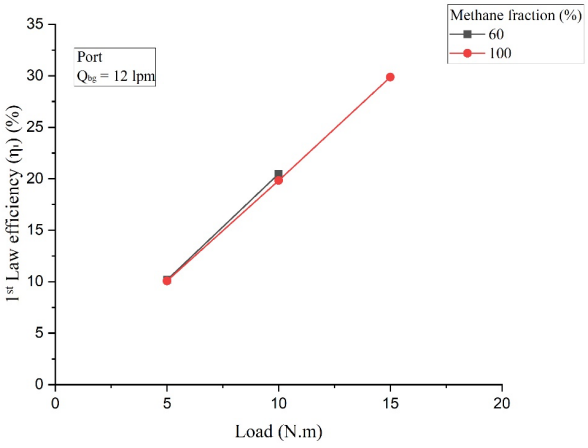

Figure 10. Brake thermal efficiency vs Load at different methane fractions.

4.3. TRENDS OF BRAKE SPECIFIC FUEL CONSUMPTION

This subsection examines how injector location, engine load, methane fraction, and biogas flow rate together influence Brake Specific Fuel Consumption (BSFC) in an internal combustion engine. Using a detailed experimental dataset and a machine learning model, the study aimed to uncover trends among these parameters and their effects on BSFC. Three injector positions were tested: port injection, Manifold 1 (6 cm from the port), and Manifold 2 (10 cm from the port). As shown in Figure 11, BSFC consistently decreased with increasing engine load across all injector locations and parameter combinations. Port injection emerged as the most efficient, showing the lowest BSFC throughout the load range, which points to better energy conversion and fuel economy. While Manifold 1 and Manifold 2 had higher BSFC, they followed the same downward trend. Interestingly, Manifold 2 often showed intermediate BSFC values despite being farther from the port, which challenges the common assumption that placing injectors closer always improves efficiency. The influence of injector location on BSFC was most noticeable at lower loads, where port injection clearly outperformed the manifold setups. At higher loads, this difference became much smaller, suggesting that injector positioning matters less as load increases. Beyond injector placement and load, the study also looked at how biogas flow rate and methane fraction affect BSFC. A higher biogas flow rate generally led to increased BSFC, though BSFC still dropped with higher loads, highlighting why it is important to carefully set the flow rate. Likewise, a higher methane fraction was linked to increased BSFC, as seen in Figures 12 and 13, which shows that the methane content needs to be adjusted based on expected load conditions for the best performance.

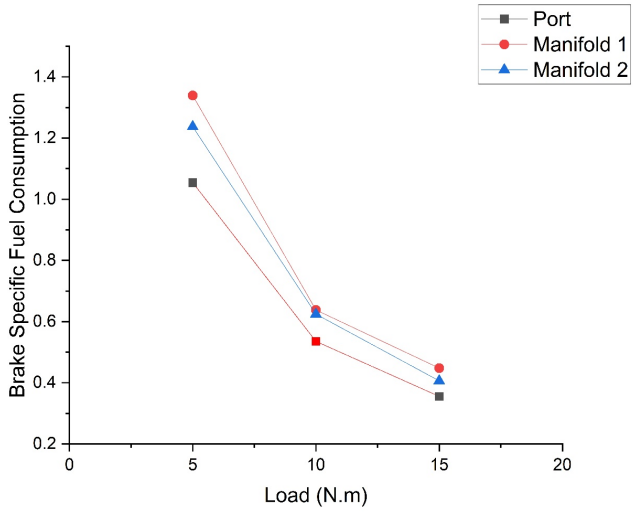

Figure 11. BSFC vs Load at all injector locations.

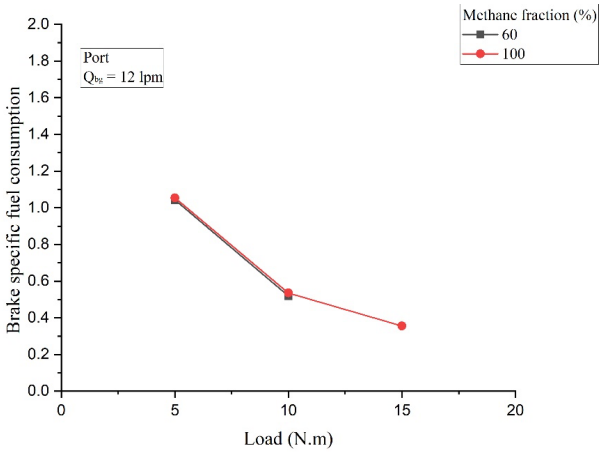

Figure 12. BSFC vs Load at different methane fractions.

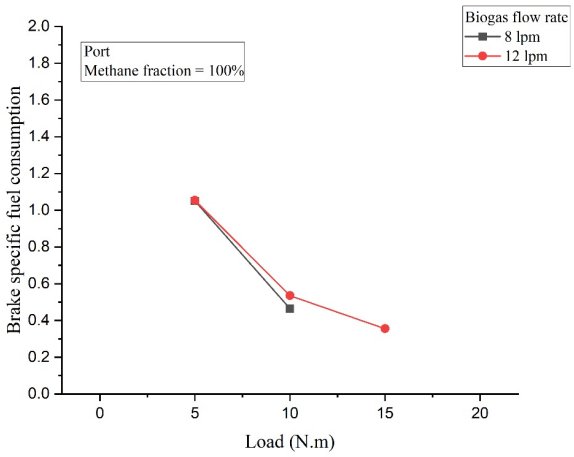

Figure 13. BSFC vs Load at different biogas flow rates.

4.4. TRENDS OF EXERGY DESTROYED

This subsection examines how injector location, engine load, methane fraction, and biogas flow rate influence exergy destruction in an internal combustion engine. Using a detailed experimental dataset, the study analyzed the relationships between these parameters to support the development of a machine learning model for optimizing engine efficiency. Three injector positions were tested: port injection, Manifold 1 (6 cm from the port), and Manifold 2 (10 cm from the port). The results, shown in the graph, plot exergy destruction against engine load for each injector location at a methane fraction of 100% and a biogas flow rate of 12 lpm. Exergy destruction decreased with increasing engine load for all injector setups. Port injection consistently showed the lowest exergy destruction, indicating more efficient energy use. Manifold 1 and Manifold 2 recorded higher values but followed the same downward trend. Notably, Manifold 1 often showed intermediate performance despite being closer to the port, challenging the assumption that closer injector placement always reduces exergy destruction. At lower loads, differences between injector locations were more pronounced, with port injection performing best as shown in Figure 14. As load increased, these differences became smaller, suggesting that injector placement is less critical at higher loads. The study also evaluated how biogas flow rate and methane fraction affect exergy destruction. Higher flow rates generally led to more exergy destruction, although values still decreased with increasing load. A higher methane fraction usually resulted in lower exergy destruction at low loads, but this trend reversed at higher loads, as seen in Figures 15 and 16.

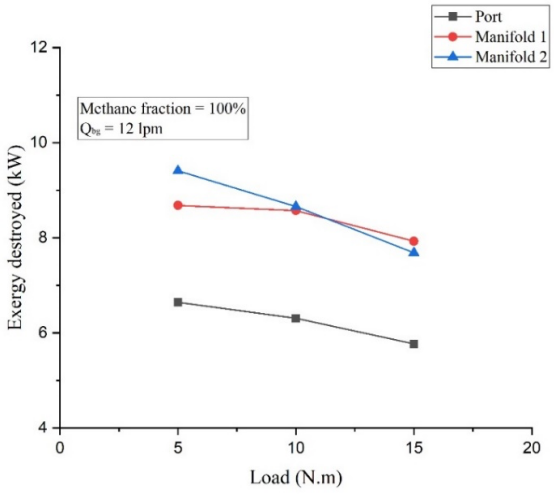

Figure 14. Exergy Destroyed vs Load at various injector locations.

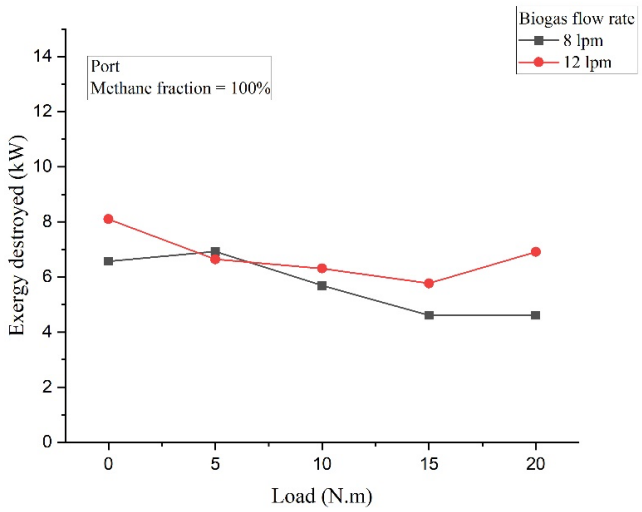

Figure 15. Exergy Destroyed vs Load at various Biogas flow rates.

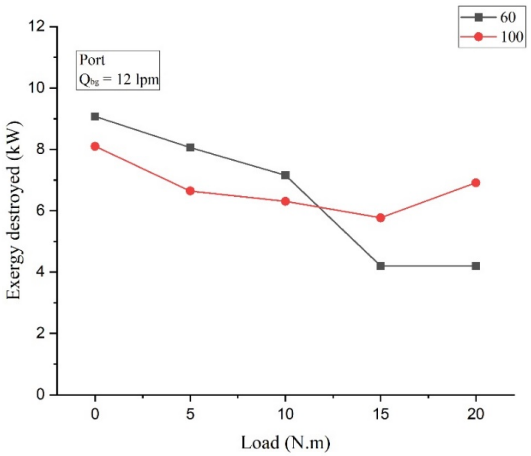

Figure 16. Exergy Destroyed vs Load at various Methane fractions.

4.5. TRENDS OF EXERGY EFFICIENCY

This section examines how injector location, engine load, methane fraction, and biogas flow rate affect exergy efficiency in the internal combustion engine. By studying both exergy destruction and efficiency, the analysis gives a clearer picture of energy use and supports building a stronger machine learning model. Tests were carried out with three injector positions: port injection, Manifold 1 (6 cm), and Manifold 2 (10 cm). The results, shown in the graph, plot exergy efficiency against engine load for a methane fraction of 100% and a biogas flow rate of 12 lpm. Exergy efficiency increased with load for all setups. Port injection consistently showed the highest efficiency across loads, as seen in Figure 17. Manifold 1 and 2 had lower efficiency but followed the same trend, with Manifold 2 often performing between the other two despite being farther from the port. At low loads, differences between injector locations were more pronounced, while at higher loads these gaps narrowed. Higher biogas flow rates generally led to lower efficiency, though efficiency still rose with load. Lower methane fractions slightly improved efficiency at higher loads, as shown in Figures 18 and 19, highlighting the need to carefully balance these factors for better engine performance.

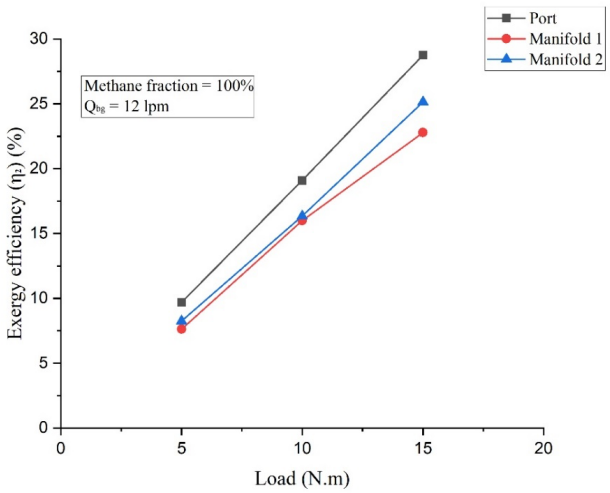

Figure 17. Exergy Efficiency vs Load at various injector locations.

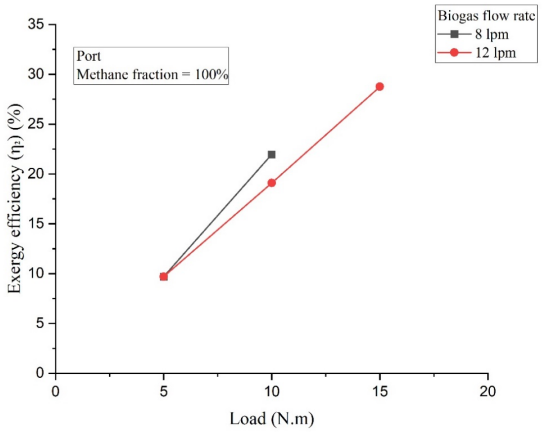

Figure 18. Exergy Efficiency vs Load at various biogas flow rates.

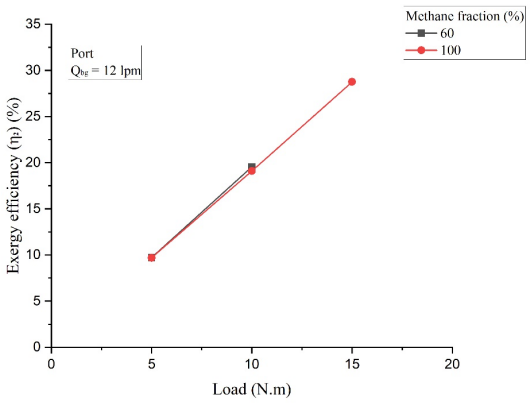

Figure 19. Exergy Efficiency vs Load at various methane fractions.

5. PSEUDOCODE AND PARAMETER SPECIFICATIONS FOR MODEL TRAINING

This section presents the pseudocode and key parameters for each machine learning model employed in this study, namely Support Vector Machine (SVM), Random Forest, AdaBoost, CATBoost, and XGBoost.

5.1. SUPPORT VECTOR MACHINE (SVM)

The first machine learning model utilized in this study was the Support Vector Machine (SVM), which identifies a decision boundary, or hyperplane, that best separates different classes by maximizing the margin between them. Thanks to this approach, SVM excels with high-dimensional data and is versatile enough to manage both linear and non-linear classification tasks through the application of various kernel functions. For predicting continuous outputs, the Support Vector Regression (SVR) variant relies on the same foundational concept. Table 1 outlines the pseudocode for this model, with key parameter configurations detailed below (Suthaharan, 2016):

Table 1. Pseudocode for the support vector machine algorithm for prediction.

Pseudocode for SVR

Input: Training dataset X with corresponding labels y and parameters: kernel type, C (regularization parameter), γ (for RBF kernel)

Output: Predicted output value for new input x_{new}

Begin

1. Define kernel function $K(x_i, x_j)$ based on the chosen kernel type (linear, polynomial, RBF, etc.).
2. Initialize regularization parameter C to control the trade-off between margin width and error.
3. For RBF kernel, set γ to determine the influence of a single training example.
4. Optimize:
   a. Calculate weights and bias by minimizing regression error.
   b. Use an optimization algorithm (e.g., Sequential Minimal Optimization) to solve for Lagrange multipliers.
   c. Identify support vectors based on non-zero Lagrange multipliers.
5. Calculate Prediction Value:

For new input x_{new} , compute:

$$\hat{y}(x_{\text{new}}) = \sum_i \alpha_i \cdot K(x_i, x_{\text{new}}) + b$$

6. Return Predicted Output Value: $\hat{y}(x_{\text{new}})$ for x_{new} .

End

Working Parameters:

- Kernel Type: Determines how data is projected into higher-dimensional space to achieve clearer separation. Common kernels include Linear (ideal for data that is linearly separable), Polynomial (captures polynomial relationships), Radial Basis Function (RBF, effective for non-linear patterns by considering distance-based similarity), and Sigmoid (comparable to neural network activation functions).
- Regularization parameter (C): Balances the goal of maximizing the margin with minimizing classification errors. Larger values of C impose stricter penalties on misclassifications, while smaller values allow for a wider margin at the expense of tolerating some errors.
- Gamma (γ): Mainly used with the RBF kernel, this parameter dictates how much influence a single training example has. Lower gamma leads to a broader, more generalized influence, whereas higher gamma concentrates the model's focus on nearby points.
- Epsilon (ϵ): Defines a tolerance zone for errors in SVR, within which no penalty is applied. This helps adjust the model's sensitivity to small deviations in data.

5.2. RANDOM FOREST

The study also incorporated the Random Forest algorithm, an ensemble learning approach that boosts prediction reliability and accuracy by constructing multiple decision trees. During training, Random Forest generates many such trees and combines their outputs taking a majority vote for classification problems or averaging (mean or median) for regression tasks. This ensemble strategy significantly enhances robustness and helps mitigate overfitting. Table 2 presents the pseudocode for the Random Forest model, while the associated working parameters are provided in the subsequent section (Breiman, 2001).

Table 2. Pseudocode for the Random Forest algorithm.

Pseudocode for Random Forest (Prediction)

Input: Trained Random Forest model, new input data x
Output: Predicted output value for $\boldsymbol{x}$

Begin

1. For each decision tree $t = 1,2, \dots, K$ in the Random Forest:
   a. Pass the input x through decision tree t to get a prediction y_t.
2. For classification, compute the mode (most common) of the predicted classes $\{y_t\}_{t=1}^{K}$. For regression, compute the mean or median of the predicted values $\{y_t\}_{t=1}^{K}$.
3. Return the final predicted output value.

End

Working Parameters of Random Forest:

- Number of Trees (K): The total trees in the forest; more trees usually improve performance but increase computation time.
- Max Depth: Limits how deep each tree can grow, preventing overfitting.
- Min Samples Split: Minimum samples required to split a node; higher values lead to simpler trees.
- Min Samples Leaf: Minimum samples in a leaf node to ensure adequate data for generalization.
- Max Features: Number of features considered for splits; adds randomness and reduces overfitting.
- Bootstrap: Indicates if samples are drawn with replacement when building trees; true adds diversity.
- Random State: Seed for reproducibility of results.
- Criterion: Measures split quality, with options for classification ('gini', 'entropy') and regression ('mse', 'mae').

5.3. ADABOOST

The third model utilized was the AdaBoost algorithm. AdaBoost is an ensemble learning method that combines multiple weak classifiers to enhance the accuracy and robustness of predictions. It operates by sequentially training classifiers, adjusting the weights of misclassified instances to focus on difficult cases. The final prediction is made by aggregating the weighted outputs of all classifiers. Table 3 shows the pseudocode for the AdaBoost algorithm, while the working parameters for the same are presented below (Schapire, 2013).

Table 3. Pseudocode for the AdaBoost algorithm.

Pseudocode for AdaBoost

Input: D, a set of d class-labeled training tuples, k, the number of rounds, a classification learning scheme.

Output: Predicted output value for tuple X.

Begin:

1. Initialize weights of tuples in D to $\frac{1}{d}$.

2. For $i = 1$ to k :
   a. Sample D with replacement based on weights to obtain D_i.
   b. Derive model M_i using D_i.
   c. Compute error $\text{error}(M_i)$.
   d. If $\text{error}(M_i) > 0.5$, go back to step 2a.
   e. Update weights of correctly classified tuples:
   i. Multiply by $\frac{\text{error}(M_i)}{1-\text{error}(M_i)}$.
   ii. Normalize weights.

To predict output for tuple :

1. Initialize weight for each class to 0.
2. For $i = 1$ to k :
   a. Compute classifier weight: $w_i = \log \frac{1-\text{error}(M_i)}{\text{error}(M_i)}$.
   b. Get prediction $y_i = M_i(X)$.
   c. Add w_i to the weight for the predicted value y_i.
3. Return the final predicted output value based on aggregated weights.

End:

Working Parameters for AdaBoost:

- Number of Rounds: This parameter defines the total number of weak classifiers to be combined. More rounds can enhance performance but may lead to overfitting.
- Weak Classifier: A basic model used in each round. Common choices include decision stumps or any classifier that performs slightly better than random guessing.
- Weight Adjustment: Each training instance has a weight that is updated after each round based on the classifier's performance. Misclassified instances receive higher weights to increase their importance in subsequent rounds.
- Error Rate: The error rate of each weak classifier is considered to adjust the weights of the classifiers in the final prediction.

5.4. CATBOOST

The fourth model employed in the study was CatBoost, a powerful gradient boosting library that is specifically designed to handle categorical features seamlessly without extensive preprocessing. CatBoost stands for Categorical Boosting and it excels at improving the performance of decision tree-based algorithms by utilizing an ordered boosting technique, which assists in preventing overfitting. One of CatBoost's key strengths is its ability to efficiently deal with a diverse array of data types, including high cardinality categorical features. Table 4 shows the pseudocode for the CATboost algorithm, while the working parameters for the same are presented below (Prokhorenkova et al., 2018).

Table 4. Pseudocode for the CATboost.

Pseudocode for CatBoost Prediction

Input: Trained CatBoost model, new input data x
Output: Predicted output value for x
Begin

1. Initialize the prediction value $f(x)$ to 0 .
2. For each decision tree $t = 1,2, \ldots,$ T in the CatBoost model:
   a. Pass the input x through decision tree t to get a prediction y_t.
   b. Update the prediction value: $f(x) = f(x) + y_t$.
3. Return the final predicted output value: $f(x)$.

End

Working Parameters for CatBoost:

- Learning Rate: Controls the step size at each iteration while moving toward a minimum of the loss function. A smaller learning rate can improve model performance but may require more iterations.
- Depth of Trees: Defines the maximum depth of the trees in the ensemble. Deeper trees can capture more complex patterns but are prone to overfitting.
- Number of Iterations: Specifies the total number of trees to be built. More iterations can enhance model performance but may also lead to overfitting if not properly managed.
- L2 Leaf Regularization: A regularization parameter that helps to prevent overfitting by penalizing large weights in the model.
- Loss Function: Determines how the model evaluates its predictions during training. Common choices include mean squared error for regression tasks and log loss for classification tasks.

5.5. XGBOOST

The final model employed in the study was XGBoost, or Extreme Gradient Boosting - a powerful machine learning algorithm that enhances the traditional gradient boosting framework. XGBoost constructs an ensemble of decision trees, where each new tree focuses on correcting the errors of its predecessors using gradient computations. By incorporating advanced regularization techniques, XGBoost effectively mitigates overfitting, a common challenge with tree-based models. The pseudocode for the XGBoost algorithm is provided in Table 5, while the key working parameters are presented below (Chen and Guestrin, 2016).

Table 5. Pseudocode for the XGBoost.

Pseudocode for XGBoost

Input: Training dataset $\{(x_i, y_i)\}_{i=1}^{n}$, learning rate η, number of trees N, loss function L
Output: Ensemble model M
Begin

1. Initialize model $M = 0$
2. For $t = 1$ to N do:
   a. Compute the residuals: $r_i = -\frac{\partial L(y_i, M(x_i))}{\partial M(x_i)}$
   b. Fit a new decision tree T_t to the residuals $\{(x_i, r_i)\}_{i=1}^{n}$
   c. Calculate the tree's contribution: $f_t(x) = \eta \cdot T_t(x)$
   d. Update the model: $M = M + f_t$
3. Return the ensemble model M

End

Working Parameters for XGBoost:

- Learning Rate: Also referred to as eta, this parameter determines how much each tree contributes to the overall prediction. Using a lower learning rate often improves the model's performance but typically requires more boosting iterations.
- Max Depth: Sets the maximum depth of the individual trees. Deeper trees are capable of capturing more intricate patterns but may also increase the risk of overfitting.
- Number of Estimators: Specifies how many trees are included in the ensemble. While increasing this number can boost model accuracy, it is important to monitor closely to avoid overfitting.
- Subsample: Indicates the fraction of the training data used to grow each tree. Values below 1 introduce randomness, which can help mitigate overfitting.
- Col sample by tree: Defines the proportion of features selected randomly for each tree, further helping to control overfitting.
- Gamma: A regularization parameter that sets the minimum reduction in loss needed to make an additional split on a leaf node, thereby managing model complexity.
- Loss Function: Determines the objective to be minimized during training. Typical choices include mean

squared error for regression and binary logistic loss for classification tasks.

6. ANALYSIS OF MACHINE LEARNING MODELS

This segment of the research evaluates the performance of five machine learning algorithms Support Vector Machine (SVM), Random Forest, AdaBoost, CATBoost, and XGBoost in correlating input variables such as injector location, methane ratio, engine load, intake air temperature, and biogas flow with essential output metrics. These outputs encompass brake specific energy consumption, exergy destruction, brake thermal efficiency, brake specific fuel consumption, and exergy efficiency. The main objective is to pinpoint the most influential set of input parameters that lead to enhanced engine efficiency and performance across different loading conditions. All models were systematically tuned using a Grid Search with ten fold cross validation to identify optimal hyperparameters and ensure robust predictive performance.

6.1. MODEL PERFORMANCES IN PREDICTING BRAKE SPECIFIC ENERGY CONSUMPTION

Brake Specific Energy Consumption (BSEC) serves as a crucial metric for assessing the energy efficiency of an engine. To predict BSEC, five machine learning models were trained using data obtained from the experimental trials and were assessed based on RMSE, MAE, and R^2 values. As illustrated in Table 6, the Random Forest algorithm delivered the best results, achieving the lowest RMSE (0.5402) and MAE (0.2879), along with the highest R^2 of 0.9501, which indicates it could account for approximately 95% of the variability in BSEC. The optimal Random Forest configuration consisted of 300 estimators with a maximum tree depth of 6, as identified through grid search optimization. XGBoost also showed strong performance with an R^2 of 0.9285 and slightly higher error metrics, while ADABoost followed with an R^2 of 0.9115. CatBoost demonstrated moderate predictive ability with an R^2 of 0.7451 and higher RMSE (1.2205) and MAE (1.1200). In contrast, the SVM model performed poorly, with the highest RMSE (3.8684), high MAE (3.4005), and a negative R^2 (-1.5609), meaning it did worse than simply predicting the mean. Because engine experiments are time-consuming and data extraction is challenging, the dataset used was relatively small. This makes it especially important to choose robust models and tune them carefully. Grid search was used to find the best hyperparameters for each algorithm to ensure the models performed as well as possible on the available data. Figure 20 shows the actual vs predicted BSEC values for the Random Forest model, clearly illustrating its high prediction accuracy.

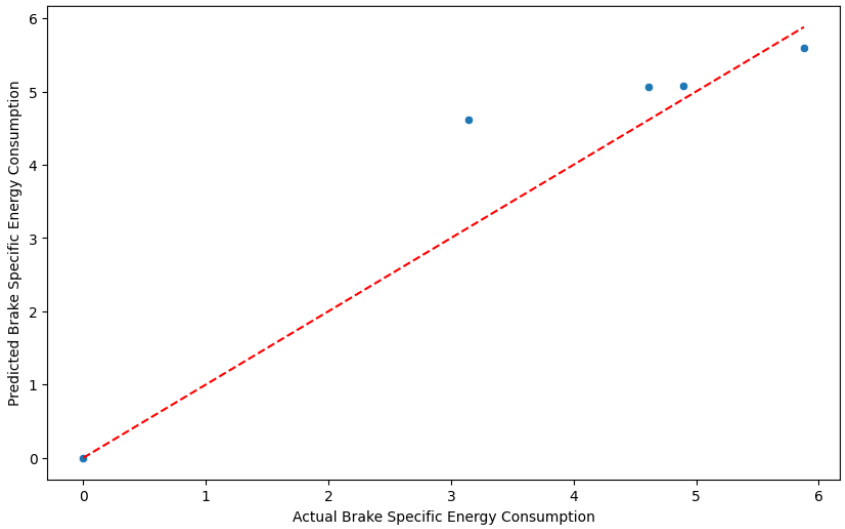

Figure 20. Actual vs Predicted plot for BSEC for the Random Forest model.

Table 6. Performance metrics of all 5 Machine Learning Models for Brake specific energy consumption Prediction.

Metric	SVM	Random Forest	ADAboost	CatBoost	XGBoost
RMSE	3.8684	0.5402	0.7192	1.2205	0.6464
MAE	3.4005	0.2879	0.4029	1.1200	0.4272

R2	-1.5609	0.9501	0.9115	0.7451	0.9285

6.2. MODEL PERFORMANCES IN PREDICTING BRAKE THERMAL EFFICIENCY

Brake Thermal Efficiency is a key parameter that influences overall engine performance and energy use. To predict this parameter, five machine learning models were tested: Support Vector Machine (SVM), Random Forest, ADABoost, CatBoost, and XGBoost. These models were trained on data from the experimental setup and evaluated using RMSE, MAE, and R^2 metrics. As shown in Table 7, the XGBoost model performed best, achieving the lowest RMSE (3.2662) and MAE (1.4464) along with the highest R^2 score of 0.9262, capturing about 93% of the variance in Brake Thermal Efficiency. The optimal XGBoost configuration obtained through grid search used 300 estimators with no restriction on maximum tree depth. Random Forest followed closely with an R^2 of 0.9136, while SVM and ADABoost also performed well with R^2 scores slightly above 0.91. CatBoost showed weaker results, with the highest RMSE (5.3956) and a lower R^2 of 0.7987. Since engine experiments are time-consuming and often produce smaller datasets, grid search was used to carefully tune model parameters and improve accuracy. Figure 21 shows the actual vs predicted plot for the XGBoost model, highlighting its ability to closely match real Brake Thermal Efficiency values. These results indicate that XGBoost, along with Random Forest, SVM, and ADABoost, can reliably predict this parameter, while CatBoost was less effective for this dataset.

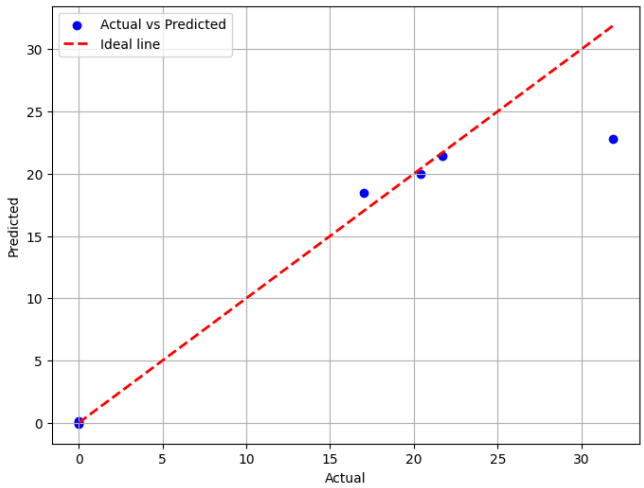

Figure 21. Actual vs Predicted plot for brake thermal efficiency for the XGboost model.

Table 7. Performance metrics of all 5 Machine Learning Models for Brake Thermal Efficiency Prediction

Metric	SVM	Random Forest	ADAboost	CatBoost	XGBoost
RMSE	3.5453	3.5345	3.6232	5.3956	3.2662
MAE	2.0964	1.5644	1.6593	4.1822	1.4464
R2	0.9131	0.9136	0.9092	0.7987	**0.9262**

6.3. MODEL PERFORMANCES IN PREDICTING BRAKE SPECIFIC FUEL CONSUMPTION

Brake Specific Fuel Consumption (BSFC) serves as an important indicator of how efficiently an engine utilizes fuel. To predict BSFC, this study employed five machine learning models Support Vector Machine (SVM), Random Forest, AdaBoost, CatBoost, and XGBoost which were trained using the experimental dataset and evaluated with RMSE, MAE, and R^2 metrics. As detailed in Table 8, XGBoost demonstrated the strongest predictive capability, achieving the lowest RMSE (0.0187) and MAE (0.0149), along with the highest R^2 value of 0.9947 The optimal XGBoost configuration obtained through grid search used 200 estimators with maximum tree depth of 6, explaining nearly 99.5% of the variation in BSFC. Random Forest also performed robustly, attaining an R^2 of 0.9501, followed by AdaBoost with an R^2 of 0.9076. CatBoost showed moderate

performance ($R^2 = 0.7451$), whereas SVM struggled significantly, yielding the highest RMSE (0.3837), highest MAE (0.3376), and a negative R^2 (-1.2345), indicating it failed to reliably capture the data patterns. Because experimental engine studies often generate relatively small datasets due to the complexity of data acquisition, grid search techniques were employed to fine-tune model parameters and enhance prediction accuracy. Figure 22 illustrates the actual versus predicted BSFC values for the XGBoost model, showing a close match to experimental data. Overall, these findings underscore that ensemble methods, particularly XGBoost, along with Random Forest and AdaBoost are highly effective for modeling BSFC, while SVM proved less suitable for this application.

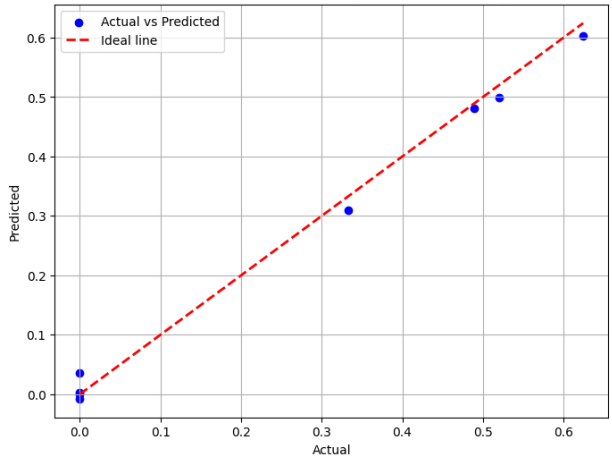

Figure 22. Actual vs Predicted plot for brake thermal efficiency for the XGboost model.

Table 8. Performance metrics of all 5 Machine Learning Models for Brake Specific Fuel Consumption Prediction

Metric	SVM	Random Forest	ADAboost	CatBoost	XGBoost
RMSE	0.3837	0.0574	0.0780	0.1296	0.0187
MAE	0.3376	0.0306	0.0445	0.1189	0.0149
R2	-1.2345	0.9501	0.9076	0.7451	**0.9947**

6.4. MODEL PERFORMANCES IN PREDICTING EXERGY DESTROYED

The Exergy Destroyed parameter, critical for assessing engine performance, was modeled using Support Vector Machine (SVM), Random Forest, ADABoost, CatBoost, and XGBoost algorithms. As shown in Table 9, the ADABoost model delivered the best results, achieving the lowest RMSE (0.5999) and MAE (0.4467), along with the highest R^2 score of 0.8818 with 100 as its number of rounds, indicating it could explain about 88% of the variance in Exergy Destroyed. XGBoost also performed well, with an RMSE of 0.7603, MAE of 0.5293, and R^2 of 0.8102. SVM and CatBoost showed moderate predictive ability, with R^2 scores of 0.7085 and 0.6494, respectively, while Random Forest performed the weakest, recording an R^2 of just 0.2698. Grid search was used to optimize hyperparameters for all models. Figure 23 illustrates the actual vs predicted values for the ADABoost model, highlighting its strong fit. These results show that ADABoost, followed by XGBoost, is best suited for predicting Exergy Destroyed with this feature set.

Table 9. Performance metrics of all 5 Machine Learning Models for Exergy Destroyed Prediction

Metric	SVM	Random Forest	ADAboost	CatBoost	XGBoost
RMSE	0.9421	1.4912	0.5999	1.0333	0.7603
MAE	0.6127	1.1602	0.4467	0.8164	0.5293
R2	0.7085	0.2698	**0.8818**	0.6494	0.8102

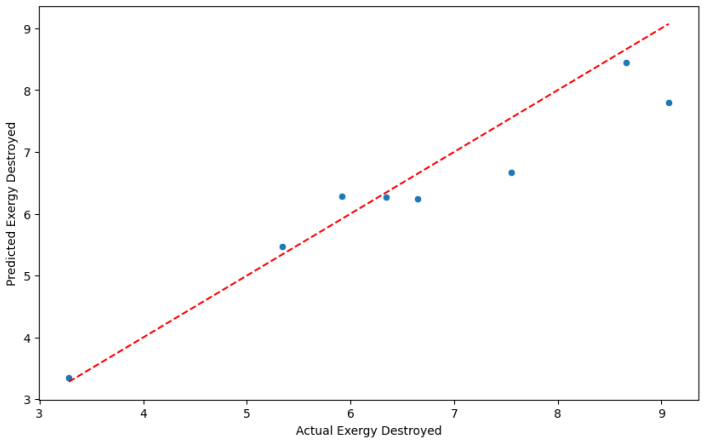

Figure 23. Actual vs predicted plot for Exergy Destroyed for the ADAboost model.

6.5. MODEL PERFORMANCES IN PREDICTING EXERGY EFFICIENCY

Exergy Efficiency, a key parameter affecting engine performance, was modeled using Support Vector Machine (SVM), Random Forest, ADABoost, CatBoost, and XGBoost. As shown in Table 10, the XGBoost model delivered the best results, achieving the lowest MAE (1.5157), an RMSE of 3.2778, and the highest R^2 score of 0.9189 with 300 estimators and maximum depth of 10, explaining about 92% of the variance in Exergy Efficiency. Random Forest, SVM, and ADABoost showed similar performance, with R^2 values between 0.9042 and 0.9156, while Random Forest also achieved a notably low MAE of 1.4805. In contrast, CatBoost performed the weakest, with an RMSE of 5.2108, MAE of 4.0571, and an R^2 of 0.7951. Grid search was used to tune hyperparameters for all models to enhance prediction accuracy. Figure 24 shows the actual vs predicted plot for XGBoost, highlighting its strong fit. Overall, XGBoost proved most effective for predicting Exergy Efficiency, with Random Forest also showing competitive results.

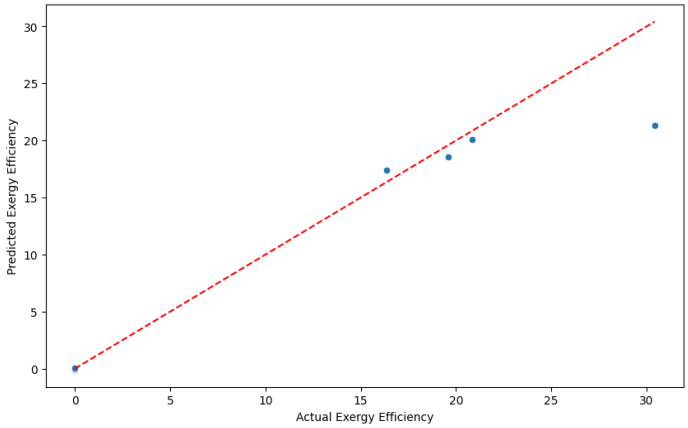

Figure 24. Actual vs predicted plot for Exergy Efficiency for the XGboost model.

Table 10. Performance metrics of all 5 Machine Learning Models for Exergy Efficiency Prediction

Metric	SVM	Random Forest	ADAboost	CatBoost	XGBoost
RMSE	3.4026	3.3451	3.5635	5.2108	3.2778
MAE	2.0518	1.4805	1.6044	4.0571	1.5157
R2	0.9126	0.9156	0.9042	0.7951	**0.9189**

6.6. COMPARISON OF ALL 5 MACHINE LEARNING MODELS FOR BEST PREDICTIVE CAPABILITIES

The performance of various machine learning models in predicting key engine parameters was evaluated

across five critical outputs: Brake Specific Energy Consumption, Brake Thermal Efficiency, Brake Specific Fuel Consumption, Exergy Destroyed, and Exergy Efficiency. As shown in Table 11, XGBoost achieved exceptional accuracy for Brake Specific Fuel Consumption with an R^2 of 0.9947, while Random Forest performed best for Brake Specific Energy Consumption at 0.9501. XGBoost also delivered strong results for Brake Thermal Efficiency and Exergy Efficiency, with R^2 scores of 0.9262 and 0.9189 respectively. For Exergy Destroyed, ADABoost showed the highest predictive performance, reaching an R^2 of 0.8818. Although the datasets were relatively small, which is typical given the time and effort required for detailed engine experiments, the models were carefully tuned using grid search to achieve the best possible results. The high R^2 values across most parameters indicate that the models effectively captured the main relationships between inputs and outputs. These relationships were often monotonic, such as efficiency generally increasing with load and specific consumptions decreasing, which is consistent with well-established engine behavior. Tree-based ensemble models such as XGBoost and Random Forest are particularly effective in capturing nonlinear interactions and threshold effects between engine operating parameters, which explains their superior performance for most outputs. Random Forest showed strong generalization for energy-based metrics due to its robustness against overfitting, while XGBoost benefited from gradient-based boosting to refine predictions for efficiency-related parameters. ADABoost performed better for exergy destruction likely because it emphasizes difficult-to-predict samples, which is useful for capturing losses influenced by subtle thermodynamic variations.

Table 11. Best performing models for prediction of each of the 5 output parameters.

Parameter	Model with highest R2 score	R2 Score
Brake specific energy Consumption	Random Forest	0.9501
Brake Thermal Efficiency	XGBoost	0.9262
Brake Specific Fuel Consumption	XGboost	0.9947
Exergy Destroyed	ADAboost	0.8818
Exergy Efficiency	XGBoost	0.9189

7. CONCLUSION

The predictive performance of multiple machine learning models was evaluated for five key engine outputs: brake specific energy consumption, brake thermal efficiency, brake specific fuel consumption, exergy destroyed, and exergy efficiency. Support Vector Machine (SVM), Random Forest, AdaBoost, CatBoost, and XGBoost models were trained and compared. As summarized in Table 11, XGBoost achieved the highest accuracy for several metrics, with R^2 values of 0.9947 for brake specific fuel consumption, 0.9262 for brake thermal efficiency, and 0.9189 for exergy efficiency. Random Forest performed best for brake specific energy consumption ($R^2 = 0.9501$), while AdaBoost yielded the highest accuracy for exergy destroyed ($R^2 = 0.8818$). The strong predictive performance indicates that the models successfully captured the dominant relationships between engine inputs and outputs, which were largely monotonic and consistent with established engine performance behavior (e.g., increased load leading to improved efficiency and reduced specific consumption). This agreement with known physical trends suggests that the models are learning meaningful system behavior rather than overfitting. Although the dataset was limited, as is typical in experimental engine studies, systematic hyperparameter tuning using grid search ensured near-optimal model performance. The consistently high R^2 values demonstrate that ensemble-based models, particularly XGBoost and Random Forest, are well suited for modeling complex thermodynamic and exergy-based engine characteristics. Beyond offline prediction, these trained models can be integrated into real-time engine monitoring frameworks by mapping readily available sensor inputs (such as load, speed, and fuel flow) to key efficiency and exergy indicators. This enables continuous estimation of otherwise difficult-to-measure parameters like exergy destruction and exergy efficiency, supporting early fault detection, performance degradation monitoring, and adaptive control strategies. Such real-time capability can help operators optimize operating conditions, reduce energy losses, and improve overall engine efficiency without requiring additional physical instrumentation.

AUTHOR CONTRIBUTIONS

Aditya Sai Samavedam: Investigation, Data Curation, Formal analysis, Writing – Original Draft, Prasshanth C. V.: Investigation, Resources, Data Curation, Validation, Manavalla Sreekanth: Methodology, Software, Formal analysis, Validation, Tamilselvan P.: Formal analysis, Methodology, Validation, Writing – Review & Editing, Feroskhan M.: Conceptualization, Methodology, Supervision, Project administration, Writing – Review & Editing.

COMPETING INTERESTS

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY STATEMENT

The data used in this study will be made available upon request.

REFERENCES

Ağbulut, Ü., Gürel, A.E. and Sarıdemir, S. (2021) 'Experimental investigation and prediction of performance and emission responses of a CI engine fuelled with different metal-oxide based nanoparticles–diesel blends using different machine learning algorithms', Energy, 215, p. 119076. https://doi.org/10.1016/j.energy.2020.119076

Balki, M.K., Çavuş, V., Duran, İ.U., Tuna, R. and Sayin, C. (2018) 'Experimental study and prediction of performance and emission in an SI engine using alternative fuel with artificial neural network', International Journal of Automotive Engineering and Technologies, 7(1), pp. 58–64. https://doi.org/10.18245/IJAET.438048

Breiman, L. (2001) 'Random forests', Machine Learning, 45(1), pp. 5–32. https://doi.org/10.1023/A:1010933404324

Çay, Y. (2013) 'Prediction of a gasoline engine performance with artificial neural network', Fuel, 111, pp. 324–331. https://doi.org/10.1016/j.fuel.2012.12.040

Çay, Y., Çiçek, A., Kara, F. and Sağiroğlu, S. (2012) 'Prediction of engine performance for an alternative fuel using artificial neural network', Applied Thermal Engineering, 37, pp. 217–225. https://doi.org/10.1016/j.applthermaleng.2011.11.019

Chen, T. and Guestrin, C. (2016) 'XGBoost: A scalable tree boosting system', in Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco, USA, pp. 785–794. https://doi.org/10.1145/2939672.2939785

Correa Gonzalez, S., et al. (2021) 'Prediction of gasoline blend ignition characteristics using machine learning models', Energy & Fuels, 35(11), pp. 9332–9340. https://doi.org/10.1021/acs.energyfuels.1c00749

Ghobadian, B., Rahimi, H., Nikbakht, A.M., Najafi, G. and Yusaf, T.F. (2009) 'Diesel engine performance and exhaust emission analysis using waste cooking biodiesel fuel with an artificial neural network', Renewable Energy, 34(4), pp. 976–982. https://doi.org/10.1016/j.renene.2008.08.008

Gökalp, B., Ertunç, H.M., Hösöz, M. and Saraç, H.İ. (2010) 'Performance prediction of a CI engine using artificial neural network for various SME and diesel fuel blends', International Journal of Vehicle Design, 54(2), pp. 156–176. https://doi.org/10.1504/IJVD.2010.035357

Hanuschkin, A., et al. (2021) 'Machine learning–based analysis of in-cylinder flow fields to predict combustion engine performance', International Journal of Engine Research, 22(1), pp. 257–272. https://doi.org/10.1177/1468087419833269

Kale, A.V. and Krishnasamy, A. (2024) 'Application of machine learning for performance prediction and optimization of a homogeneous charge compression ignited engine operated using biofuel–gasoline blends', Energy Conversion and Management, 314, p. 118629. https://doi.org/10.1016/j.enconman.2024.118629

Katreddi, S. (2023) Development of a machine learning-based approach to predict fuel consumption and maintenance cost of heavy-duty vehicles using diesel and alternative fuels. PhD thesis. West Virginia University. Available at: https://researchrepository.wvu.edu/etd/11780

(Accessed: 18 June 2024). https://doi.org/10.33915/etd.11780

Kullolli, S., Sakthivel, G. and Ilangkumaran, M. (2016) 'A neural network model for the prediction of compression ignition engine performance at different injection timings', International Journal of Ambient Energy, 37(3), pp. 227–236. https://doi.org/10.1080/01430750.2014.931297

Leo, G.M.L., Jayabal, R., Srinivasan, D., Das, M.C., Ganesh, M. and Gavaskar, T. (2024) 'Predicting the performance and emissions of an HCCI-DI engine powered by waste cooking oil biodiesel with nano additives and gasoline injection using random forest', Fuel, 357, p. 129914. https://doi.org/10.1016/j.fuel.2023.129914

Liu, J., Ulishney, C. and Dumitrescu, C.E. (2020) 'Improving machine learning model performance in predicting the indicated mean effective pressure of a natural gas engine', in ASME Internal Combustion Engine Division Fall Technical Conference (ICEF 2020). https://doi.org/10.1115/ICEF2020-2968

Liu, J., Ulishney, C. and Dumitrescu, C.E. (2021) 'Random forest machine learning model for predicting combustion feedback information of a natural gas spark ignition engine', Journal of Energy Resources Technology, 143(1). https://doi.org/10.1115/1.4047761

Liu, Y., et al. (2022) 'Machine learning based predictive modelling of a micro gas turbine engine fuelled with microalgae blends using LSTM networks', Fuel, 322, p. 124183. https://doi.org/10.1016/j.fuel.2022.124183

Liu, Z. and Karimi, I.A. (2020) 'Gas turbine performance prediction via machine learning', Energy, 192, p. 116627. https://doi.org/10.1016/j.energy.2019.116627

Liu, Z. and Liu, J. (2022) 'Machine learning assisted analysis of an ammonia engine performance', Journal of Energy Resources Technology, 144(11). https://doi.org/10.1115/1.4054287

Makalesi, A., et al. (2022) 'Using machine learning methods to predict the effect of alternative fuel mixtures on exhaust emissions', European Journal of Science and Technology, 34, pp. 273–279. https://doi.org/10.31590/EJOSAT.1081539

Mohammad, A., Rezaei, R., Hayduk, C., Delebinski, T.O., Shahpouri, S. and Shahbakhti, M. (2021) 'Hybrid physical and machine learning-oriented modeling approach to predict emissions in a diesel compression ignition engine', SAE Technical Papers. https://doi.org/10.4271/2021-01-0496

Oğuz, H., Saritas, I. and Baydan, H.E. (2010) 'Prediction of diesel engine performance using biofuels with artificial neural network', Expert Systems with Applications, 37(9), pp. 6579–6586. https://doi.org/10.1016/j.eswa.2010.02.128

Patel, S., Azad, A.K. and Khan, M. (2019) 'Numerical investigation for predicting diesel engine performance and emission using different fuels', Energy Procedia, 160, pp. 834–841. https://doi.org/10.1016/j.egypro.2019.02.150

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A.V. and Gulin, A. (2018) 'CatBoost: Unbiased boosting with categorical features', in Advances in Neural Information Processing Systems (NeurIPS 2018). Available at: proceedings.neurips.cc (Accessed: 29 October 2024). https://doi.org/10.48550/arXiv.1706.09516

Samavedam, A.S., C.V., P., Sreekanth, M., Khan, T.M.Y., Almakayeel, N. and M., F. (2024) 'Exergetic investigation of the influence of injector location in HCCI engines utilizing DEE as pilot fuel and biogas as primary fuel', Heliyon, 10(18). https://doi.org/10.1016/j.heliyon.2024.e37607

Sahoo, S., Kumar, V.N.S.P. and Srivastava, D.K. (2022) 'Quantitative analysis of engine parameters of a variable compression ratio CNG engine using machine learning', Fuel, 311, p. 122587. https://doi.org/10.1016/j.fuel.2021.122587

Schapire, R.E. (2013) 'Explaining AdaBoost', in Empirical inference: Festschrift in honor of Vladimir N. Vapnik. Berlin: Springer, pp. 37–52. https://doi.org/10.1007/978-3-642-41136-6_5

Sebayang, A.H., et al. (2017) 'Prediction of engine performance and emissions with Manihot glaziovii bioethanol–gasoline blends using extreme learning machine', Fuel, 210, pp. 914–921. https://doi.org/10.1016/j.fuel.2017.08.102

Sebayang, A.H., et al. (2022) 'Modelling and prediction approach for engine performance and exhaust emission based on artificial intelligence of Sterculia foetida biodiesel', Energy Reports, 8, pp. 8333–8345. https://doi.org/10.1016/j.egyr.2022.06.052

Suthaharan, S. (2016) 'Support vector machine', in Machine learning models and algorithms for big data classification. Boston: Springer, pp. 207–235. https://doi.org/10.1007/978-1-4899-7641-3_9

Wang, H., et al. (2022) 'Comparison and evaluation of advanced machine learning methods for performance and emissions prediction of a gasoline Wankel rotary engine', Energy, 248, p. 123611. https://doi.org/10.1016/j.energy.2022.123611

Yang, R., et al. (2022) 'An artificial neural network model to predict efficiency and emissions of a gasoline engine', Processes, 10(2), p. 204. https://doi.org/10.3390/pr10020204

Zandie, M., Ng, H.K., Gan, S., Muhamad Said, M.F. and Cheng, X. (2023) 'Multi-input multi-output machine learning predictive model for engine performance, emissions, combustion and ignition characteristics of diesel–biodiesel–gasoline blends', Energy, 262, p. 125425. https://doi.org/10.1016/j.energy.2022.125425