applied sciences

MDPI

Article

Research on Critical Quality Feature Recognition and Quality Prediction Method of Machining Based on Information Entropy and XGBoost Hyperparameter Optimization

Dongyue Qu [1], Chaoyun Gu [1], Hao Zhang [2], Wenchao Liang [1], Yuting Zhang [2] and Yong Zhan [1,*]

[1] College of Mechanical and Electrical Engineering, Harbin Engineering University, Harbin 150001, China; qudongyue@hrbeu.edu.cn (D.Q.); chaoyungu@hrbeu.edu.cn (C.G.); liangwebcgao@hrbeu.edu.cn (W.L.)
[2] Institute of Advanced Technology, Heilongjiang Academy of Sciences, Harbin 150080, China; a18686795549@163.com (H.Z.); yutingzhanghas@126.com (Y.Z.)
* Correspondence: zhanyong@hrbeu.edu.cn

Abstract: To address the problem of predicting machining quality for critical features in the manufacturing process of mechanical products, a method that combines information entropy and XGBoost (version 2.1.1) hyperparameter optimization is proposed. Initially, machining data of mechanical products are analyzed based on information entropy theory to identify critical quality characteristics. Subsequently, a quality prediction model for these critical features is established using the XGBoost machine learning framework. The model's hyperparameters are then optimized through Bayesian optimization. This method is applied as a case study to a medium-speed marine diesel engine piston. After the critical quality characteristics in the machining process are identified, the machining quality of these vital characteristics is predicted, and the results are compared with those obtained from a machine learning model without hyperparameter optimization. The findings demonstrate that the proposed method effectively predicts the machining quality of mechanical products.

Keywords: critical quality characteristics; machining quality prediction; information entropy; XGBoost; Bayesian optimization

Citation: Qu, D.; Gu, C.; Zhang, H.; Liang, W.; Zhang, Y.; Zhan, Y. Research on Critical Quality Feature Recognition and Quality Prediction Method of Machining Based on Information Entropy and XGBoost Hyperparameter Optimization. *Appl. Sci.* **2024**, *14*, 8317. https://doi.org/10.3390/app14188317

Academic Editor: Luis Javier Garcia Villalba

Received: 13 August 2024
Revised: 5 September 2024
Accepted: 13 September 2024
Published: 15 September 2024

1. Introduction

In the modern manufacturing industry, the quality of mechanical products is of paramount importance. The manufacturing of these products involves complex processes that demand stringent control measures to ensure accuracy and consistency throughout production. Critical quality characteristics are particularly significant, as they directly influence the final product's performance, reliability, and lifespan [1]. Consequently, accurately predicting the processing quality of these critical characteristics is vital for maintaining a competitive edge and adhering to industry standards.

In recent years, scholars have extensively explored the identification of key quality features and the prediction of processing quality in machining. The machining process of mechanical parts typically involves multiple steps, each targeting different quality characteristics. It is essential to identify the most significant quality characteristics based on specific evaluation criteria to enable in-depth analysis in subsequent studies. The Analytic Hierarchy Process (AHP), developed by Thomas L. Saaty in 1971, is a multi-criteria decision-making method that assists decision makers in ranking the pros and cons of different options by hierarchically decomposing complex problems [2]. For example, Deng et al. [3] applied AHP to identify key influencing factors in the complex deformation of flexible materials during processing. They developed a model with importance as the target layer, processing attributes as the criterion layer, and deformation influencing factors as the index layer, systematically analyzing the relative importance of each factor. Zhang et al. [4] introduced the fuzzy consistent matrix and fuzzy analytic hierarchy process (FAHP), which, by incorporating a fuzzy function, enhances the consistency of the judgment matrix in the traditional AHP, thereby reducing the impact of subjective factors and uncertainty. Additionally, the causal matrix method identifies key influencing factors based on the degree of influence that input variables have on output variables [5]. Zhang utilized the causal matrix method to evaluate the manufacturing process of conical picks, treating each process as an input and the critical quality characteristics as outputs [6]. However, current analysis methods, such as AHP, FAHP, and the causal matrix method, are significantly affected by subjective factors, highlighting the urgent need for a quantitative approach to enhance the objectivity and consistency of the analysis.

Shannon introduced information entropy in information theory, originally applying it to communication to measure the average uncertainty of discrete random variables. When calculated using a base-2 logarithm, the unit of information entropy is expressed in bits [7]. Qu et al. [8] applied information theory to establish a reliability analysis model for the machining process, where the uncertainty of machining quality is assessed using vertex entropy, and the error transfer effect is quantified by the mutual information transfer coefficient. Wang et al. [9] developed a "stability entropy" method based on information entropy, which enables real-time calculation of cutting stability through spindle load data. Additionally, Zhang et al. [10] proposed a method based on information transfer entropy that achieved early fault warning and root cause tracking of CNC lathes using net entropy calculation and a sliding window method. While information entropy theory has been widely employed to analyze error transmission, evaluate CNC machining stability, and diagnose faults, there is a noticeable gap in research regarding the identification of crucial quality features in machining. To address this gap, the present study introduces information entropy as a tool for quantitatively screening crucial quality features.

Machining quality prediction aims to evaluate potential quality issues during machining, identify process defects in advance, and implement early warning measures to enhance product quality and reliability. However, due to the numerous features involved in machining, comprehensively predicting each quality feature incurs high computational costs. To address this, the present study focuses on analyzing key machining quality features based on information entropy and predicting these features to achieve a balance between efficiency and accuracy. Common machining methods for parts include turning, milling, drilling, and grinding. For example, Zajac et al. [11] successfully predicted tool performance during turning by conducting experiments on C45 material using cutting tools with various materials and coatings, employing the Taylor model and least squares method. Makhfi et al. [12] evaluated eight machine learning models to assess their performance in predicting the cutting force for AISI 52100 bearing steel hard turning through five-fold cross-validation. Their findings revealed that Gaussian process regression (GPR) and decision tree regression models performed best, with GPR also providing prediction uncertainty, thereby aiding in optimizing cutting parameters and reducing tool wear. Additionally, Alajmi and Almeshal [13] utilized the ANFIS-QPSO method to successfully predict and optimize the surface roughness of AISI 304 stainless steel during turning, showcasing the method's high accuracy and robustness. Nonetheless, each processing quality prediction method has its own strengths and limitations, making it essential to select the most suitable method to ensure the accuracy and applicability of the prediction results.

Common quality prediction methods include time series prediction [14], regression analysis, the Taylor model [11], and machine learning techniques [15]. Time series prediction, while valuable, encounters practical challenges such as difficulties in addressing nonlinear relationships, high data requirements, and the need for long, complete historical datasets, which are often difficult to obtain [16]. Regression analysis is straightforward and easy to implement, but when dealing with a large number of explanatory variables, the model can easily suffer from overfitting or underfitting issues [17]. The Taylor model is frequently used to predict tool life and cutting force, yet it struggles to cope with complex or dynamic machining environments due to its limited scope, lack of dynamic response, and reliance on experimental data. On the other hand, machine learning, a branch of artificial intelligence, excels at handling complex tasks by uncovering hidden patterns and associations within datasets [18]. Popular machine learning methods for quality prediction include decision trees [19], random forests [20], support vector machines (SVMs) [21], and XGBoost [22]. For example, Ahmed et al. [23] applied the decision tree algorithm to analyze resistance spot welding (RSW) data from an automobile manufacturer, constructing a regression tree and extracting decision rules to predict nugget width, which highlighted the influence of design and process parameters. Similarly, Ye et al. [24] employed the weighted random forest (WRF) algorithm to develop a slab quality prediction model based on multiple process parameters, effectively addressing sample imbalance in the continuous casting process and validating the method's effectiveness through real-time data. Furthermore, Yu et al. [25] proposed a method to obtain contour error of tool rotation via a workpiece shape-tool contour mapping identification test. They optimized the least squares support vector machine (LS-SVM) model using a genetic algorithm (GA), demonstrating that the GA-LS-SVM model outperformed the unoptimized LS-SVM model in error prediction. Among these methods, XGBoost stands out for its ability to handle complex nonlinear relationships, automatically select features, and perform efficiently and accurately with large-scale datasets [26]. In this study, the XGBoost method is employed to predict and analyze machining quality, with the goal of improving prediction model accuracy and reliability, thereby supporting quality control and optimization in actual production.

XGBoost captures complex data patterns through the gradient boosting framework and includes built-in functions such as missing value processing, parallel processing, and regularization, which effectively prevent overfitting and enhance generalization capability. Additionally, XGBoost's automatic feature selection and hyperparameter optimization further contribute to improving the model's performance. For these reasons, this study selects XGBoost to predict the quality of crucial processing features based on information entropy analysis, aiming to achieve efficient and accurate predictions, early warning, and enhanced product processing quality and reliability. Liao et al. [27], for instance, proposed an XGBoost load forecasting model based on similar days by analyzing the influence of meteorology and day type, using regularization terms to control complexity and prevent overfitting. Their simulation results demonstrated the model's effectiveness in short-term load forecasting. However, the performance of the XGBoost model is significantly influenced by hyperparameter settings. Without proper hyperparameter optimization, the model may experience poor performance, including reduced prediction accuracy, severe overfitting or underfitting, and extended training times [28]. Hyperparameter optimization is essential to balance model complexity and generalization capability, thereby enhancing prediction effectiveness and computational efficiency. Therefore, optimizing hyperparameters is critical to fully leverage XGBoost's strengths in quality prediction [29]. Standard methods of hyperparameter optimization include grid search and random search. Grid search exhaustively explores a predefined range of parameters to find the optimal combination, which is suitable for small datasets but computationally expensive [30]. In contrast, random search explores a broader range of combinations by randomly selecting hyperparameter values within predefined iterations, though it may struggle to find the global optimal solution. Bayesian optimization, based on Bayes' theorem, can identify the global optimal solution for complex objective functions with fewer sampling iterations. For example, Xiong et al. [31] proposed an XGBoost algorithm based on Bayesian optimization (BH-XGBoost) for short-term wind power prediction in wind farms. Compared with XGBoost, SVM, KELM, and LSTM, BH-XGBoost demonstrated higher prediction accuracy under various conditions, especially in extreme weather and low wind speed situations, highlighting its significant advantages. Despite the proven effectiveness of combining Bayesian optimization with XGBoost in other fields, there is limited research on its application in predicting crucial quality features in machining. Therefore, this study aims to fill this gap by applying this combined approach to enhance the accuracy and reliability of quality predictions in machining processes.

This study focuses on identifying key quality features in the machining of mechanical products and predicting their processing quality. As the manufacturing industry advances, the demand for higher product quality has significantly increased, particularly in the machining of parts with complex structures and multiple processes. Accurate identification and prediction of these key quality features are crucial for ensuring the performance and reliability of the final product. Traditional quality control methods often struggle to manage the complexities of modern machining environments, underscoring the need for more advanced prediction techniques.

In the second chapter, the concept of information entropy is introduced to quantitatively screen key quality features, effectively addressing the uncertainty introduced by subjective factors in traditional methods. The third chapter applies the XGBoost machine learning framework to predict the processing quality of these key features. A prediction model is developed, and Bayesian optimization is utilized to fine-tune the model's hyperparameters, thereby enhancing both the model's efficiency and prediction accuracy.

The fourth chapter validates the effectiveness of the proposed method through a case study involving a medium-speed marine diesel engine piston. In the fifth chapter, the case study results are analyzed and discussed, with potential directions for future research being explored. Finally, the sixth chapter summarizes the study, highlighting its contributions to scientific applications and the broader field of quality control in manufacturing.

2. Identification of Machining Key Quality Features Based on Information Entropy

In this study, information entropy is introduced to achieve quantitative screening of crucial quality features. It measures the average uncertainty of discrete random variables. When the base of the logarithm is 2, the unit of information entropy is a bit (bit) [8]. Information entropy is a function of the distribution of a random variable X, which depends on its probability distribution rather than any particular observation of X [32]. The more dispersed the probability distribution of the random variable, the higher its uncertainty and, consequently, the greater its information entropy. The entropy $\mathrm{H}(X)$ of any random variable X is defined by the following Formula (1):

$$\mathrm{H}(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x) \tag{1}$$

where $\mathcal{X}$ is the value space of X, and x is the observed value of X.

Random variables represent the outcomes of random tests. Before the machining process concludes, the quality of each feature is uncertain. Thus, the machining quality for each feature can be considered a discrete random variable. Critical quality features are determined by calculating the entropy of these random variables. Entropy measures the uncertainty of a random variable [32]. A smaller entropy value indicates lower uncertainty of the discrete random variable and better processing consistency of the quality characteristics related to the random variable.

3. Machining Quality Prediction Method Based on XGBoost Hyperparameter Optimization

Establishing a processing quality prediction model for key quality characteristics allows for evaluating processing quality before production, thereby reducing economic losses caused by substandard processing quality. This study utilized the eXtreme Gradient Boosting (XGBoost) machine learning framework to develop the prediction model. Bayesian optimization is employed to optimize the hyperparameters during the model's construction.

3.1. XGBoost Algorithm

Ensemble learning has become a prominent research direction in machine learning in recent years. Its core concept is to combine multiple weak learners to leverage the strengths of each. The XGBoost algorithm is a type of ensemble learning. It consists of multiple Classification and Regression Tree (CART) decision trees, where each tree predicts the difference between the current predicted value and the actual value. The predictions from each decision tree are accumulated to obtain the final predicted value. The XGBoost modeling process is illustrated in Figure 1.

$$D = \{(x_1, y_1), (x_2, y_2), L, (x_n, y_n)\} \quad (2)$$

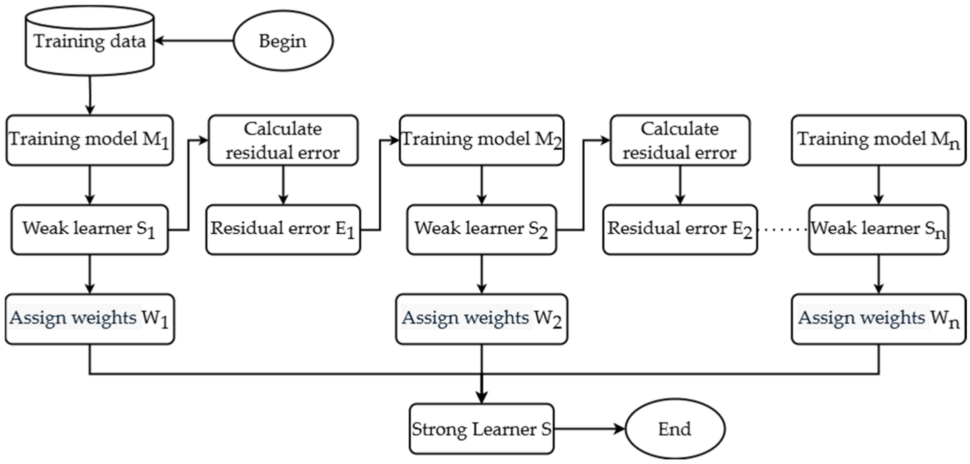

Figure 1. XGBoost modeling process.

Suppose the training dataset is denoted as D, defined by (2), where $x_i \in \mathbb{R}^m$ and $y_i \in \mathbb{R}$. Here, x_i is an M-dimensional vector representing the input features, and y_i is the sample's label. If the XGBoost model contains k regression trees, the model can be expressed as shown in the following Formula (3):

$$\hat{y}_i^{(k)} = \hat{y}_i^{(k-1)}(x_i) + f_k(x_i) \quad (3)$$

where $f_k(x)$ represents the k-th regression tree, which can be abbreviated as f_k in subsequent formulas. The regression tree filters the features of the samples, making each sample fall into the corresponding regression tree leaf nodes based on the feature filtering results. Each leaf node of the regression tree has a weight value ω, representing the leaf node's predicted value.

Like other machine learning algorithms, XGBoost also has an objective function. By minimizing the objective function, the optimal parameters of the machine learning model can be obtained. The objective function of XGBoost is defined by the following Formula (4):

$$F_{obj}^{(k)} = \sum_{i=1}^{n} L(y_i, \hat{y}_i^{(k)}) + \sum_{i=1}^{k} \Omega(f_i) \quad (4)$$

where $\sum_{i=1}^{n} L(y_i, \hat{y}_i^{(k)})$ is the sum of the loss function (the cost function), used to evaluate the degree of conformity between the predicted value and the actual value, and $\sum_{i=1}^{k} \Omega(f_i)$ is the sum of regularization terms, representing the model's complexity. The larger this value is, the more complex the model becomes, and the more prone it is to overfitting. The definition of regularization is shown in the following Equation (5), where the first term controls the complexity of the regression tree, and the second term controls the weight value ω of the leaf nodes of the regression tree:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \| \omega \|^2 \quad (5)$$

where T represents the number of leaf nodes, and γ and λ are coefficients used to balance the proportion of the first and second terms.

In solving the objective function with the minimum value as the criterion, the objective function shown in Formula (4) is challenging to solve in Euclidean space. Therefore, XGBoost uses the Taylor series expansion to approximate the objective function. By substituting Formula (3) into Formula (4), the following can be obtained:

$$\begin{aligned} F_{obj}^{(k)} &= \sum_{i=1}^{n}\left[L(y_i, \hat{y}_i^{(k-1)} + f_k(x_i))\right] + \sum_{i=1}^{k-1}\Omega(f_i) + \Omega(f_k) \\ &= \sum_{i=1}^{n}\left[L(y_i, \hat{y}_i^{(k-1)} + f_k(x_i))\right] + \Omega(f_k) + C_1 \end{aligned} \tag{6}$$

where f_k is the predicted value of the output of the k-th round, $\hat{y}_i^{(k-1)}$ is the prediction of the model for data sample i in the k-th round, $L(y_i, \hat{y}_i^{(k-1)} + f_k(x_i))$ is the error between the predicted value and the actual value of data sample i in the k-th round, and $C_1 = \sum_{i=1}^{k-1}\Omega(f_i)$ are constants.

To meet the accuracy requirements, the second-order Taylor series expansion is used to approximate the original objective function as follows:

$$F_{obj}^{(k)} = \sum\left[L(y_i, \hat{y}_i^{(k-1)}) + \partial_{\hat{y}_i^{(k-1)}} L(y_i, \hat{y}_i^{(k-1)}) \cdot f_k(x_i) + \frac{1}{2}\partial^2_{\hat{y}_i^{(k-1)}} L(y_i, \hat{y}_i^{(k-1)}) \cdot f_k^2(x_i) + \Omega(f_k)\right] + C_1 \tag{7}$$

where $\partial_{\hat{y}_i^{(k-1)}} L(y_i, \hat{y}_i^{(k-1)})$ and $\partial^2_{\hat{y}_i^{(k-1)}} L(y_i, \hat{y}_i^{(k-1)})$ represent the first and second derivatives of the loss function with respect to the model, respectively.

Let $g_i = \frac{\partial L(y_i, \hat{y}^{(k-1)})}{\partial \hat{y}^{(k-1)}}$ and $h_i = \frac{\partial^2 L(y_i, \hat{y}^{(k-1)})}{\partial \hat{y}^{(k-1)}}$ be substituted into Formula (7), where $L(y_i, \hat{y}_i^{(k-1)})$ is a constant, and the constant has no effect on the solution of the objective function. Therefore, Formula (7) can be rewritten as follows:

$$F_{obj}^{(k)} = \sum_{i=1}^{n}\left[g_i f_k(x_i) + \frac{1}{2}h_i f_k^2(x_i)\right] + \Omega(f_k) \tag{8}$$

Bringing Formula (5) into Formula (8) can obtain the following:

$$\begin{aligned} F_{obj}^{(k)} &= \sum_{i=1}^{n}\left[g_i f_k(x_i) + \frac{1}{2}h_i f_k^2(x_i)\right] + \gamma T + \frac{1}{2}\lambda \| \omega \|^2 \\ &= \sum_{i=1}^{n}\left[g_i f_k(x_i) + \frac{1}{2}h_i f_k^2(x_i)\right] + \gamma T + \frac{1}{2}\lambda \sum_{i=1}^{T} \omega_i^2 \end{aligned} \tag{9}$$

For a regression tree model, each input sample is assigned to a specific leaf node of the tree, where it is associated with a weight value ω of that leaf node. Consequently, the regression tree model $f_k(x_i)$ can be represented by the weight value ω_j of the corresponding leaf node j. The sample set assigned to leaf node j is defined as I_j, which satisfies Formula (10).

$$I_j = \{i | f_k(x_i) = j\} \tag{10}$$

Bringing ω_j and (10) into Formula (9), we can obtain the following:

$$F_{obj}^{(k)} = \sum_{j=1}^{T}\left[\left(\sum_{i \in I_j} g_i\right)\omega_j + \frac{1}{2}\left(\sum_{i \in I_j} h_i + \lambda\right)\omega_j^2\right] + \gamma T \tag{11}$$

Formula (11) can be regarded as a quadratic function with ω_j as the independent variable, assuming that the quadratic function has a minimum value when $\omega_j = \omega_j^*$. For the convenience of solving ω_j^*, let $G_j = \sum_{i \in I_j} g_i$ and $H_j = \sum_{i \in I_j} h_i$.

Bring G_j and H_j into the following Formula (11):

$$F_{obj}^{(k)} = \sum_{j=1}^{T}\left[(G_j)\omega_j + \frac{1}{2}(H_j+\lambda)\omega_j^2\right] + \gamma T \tag{12}$$

According to the formula of the most value of the quadratic function, ω_j^* can be obtained.

$$\omega_j^* = \frac{G_j}{-2 \times \frac{1}{2}(H_j+\lambda)} = -\frac{G_j}{H_j+\lambda} \tag{13}$$

The minimum of the objective function $F_{obj}^{(k)}$ is as follows:

$$F_{obj}^{(k)} = -\frac{1}{2}\sum_{j=1}^{T}\frac{G_j^2}{H_j+\lambda} + \gamma T \tag{14}$$

Equation (14) can be used to evaluate the performance of the regression tree model. A smaller value of $F_{obj}^{(k)}$ indicates better performance of the model. During the training process, Equation (14) is used to assess various regression tree models, allowing for the selection of the optimal model.

3.2. Bayesian Optimization of Hyperparameters

In the XGBoost model, hyperparameter settings directly impact the model's predictive performance. Bayesian optimization (BO) is a sequential model optimization method used for black-box function optimization. The pseudocode for the Bayesian optimization algorithm is shown in Table 1. It primarily consists of two core components: the probabilistic surrogate model and the acquisition function. The surrogate model fits the objective function and is recursively estimated based on the known data-optimal value [33]. Common surrogate models include the Gaussian process, random forest, Beta–Bernoulli model, etc. Among these, the Gaussian process has a significant advantage due to its integration of parameter uncertainty [34]. Therefore, this study employs the Gaussian process as the surrogate model for Bayesian optimization. A Gaussian process is characterized by its mean function $\mu(x)$ and covariance function $k(x, x')$, as follows:

$$f(x) \sim GP\big(\mu(x), k(x, x')\big) \tag{15}$$

where the mean function $\mu(x)$ satisfies $\mu(x) = \mathrm{E}[q(x)]$, and the covariance function $k(x, x')$ satisfies $k(x, x') = \mathrm{E}[(q(x) - \mu(x))(q(x') - \mu(x'))]$.

Table 1. Bayesian optimization pseudocode.

Input:	**The Initial Sample Number N, the Maximum Number of Iterations T, the Decision Space Ψ, and the Optimal Sampling Point x^* Are Initialized.**
Step 1	In the decision space Ψ, N sample points are extracted according to uniform distribution.
Step 2	The objective function value is obtained by evaluating the initial sample points, and the dataset ζ is constructed to obtain the current optimal $\boldsymbol{x}^*$.
Step 3	The Gaussian surrogate model GP is constructed.
Step 4	For $t = 1, 2, 3, \cdots, T$ do
Step 5	The maximum value of the acquisition function is calculated, and a new sampling point is obtained at the maximum value.
Step 6	Evaluate the new sampling point, update the dataset ζ, and update the optimal sampling point $\boldsymbol{x}^*$.
Step 7	Update the Gaussian surrogate model GP.
Step 8	End For
output:	Optimal sampling point $\boldsymbol{x}^*$
input:	The initial sample number N, the maximum number of iterations T, the decision space Ψ, and the optimal sampling point $\boldsymbol{x}^*$ are initialized.

In practical engineering applications, it is often challenging to provide a highly reasonable prior mean function, so $\mu(x)$ is frequently set to 0 [34]. The covariance function characterizes the degree of correlation between two points in the decision space. The norm is commonly used to measure the distance between two points. A smaller norm between two points indicates a higher correlation and a larger value of the covariance function, which satisfies the following Equation (16):

$$k(x,x_1) > k(x,x_2), if||x-x_1||<||x-x_2|| \tag{16}$$

Selecting an appropriate covariance function is crucial. Common stationary covariance functions include the Matérn, exponential, and Gaussian kernel functions. This paper chooses the Matérn function for its flexibility [35], as shown in Equation (17). The hyperparameters in the following Equation (17) are optimized by maximizing the marginal likelihood [35]:

$$k(x,x') = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}r}{l}\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu}r}{l}\right) \tag{17}$$

where $\Gamma(\cdot)$ is the Γ function; $r = |x - x'|$; ν, l is a constant greater than 0; K_ν is the improved Bessel function.

In iterative Gaussian process regression, the actual function value $q(\boldsymbol{x})$ of the input $\boldsymbol{x}$ can be theoretically calculated. However, in practical applications, $q(\boldsymbol{x})$ is often not calculated but observed. The observed data typically contain noise, represented by $y_n = q(\boldsymbol{x}) + \epsilon$ ($\epsilon \sim N(0, \sigma^2_{noise})$), where the noise ϵ epsilonϵ is assumed to be independently and identically distributed.

Common acquisition functions include the probability improvement (PI) acquisition function, the expectation improvement (EI) acquisition function, and the upper confidence boundary (UCB) acquisition function [36]. This study selects the expectation improvement acquisition function $\alpha_{\mathrm{EI}}(\boldsymbol{x};\zeta)$, whose mathematical expression is shown below. The observation formula indicates that the expectation improvement acquisition function can provide both the probability of improvement from sampling at new points and the magnitude of the improvement obtained, as follows:

$$\alpha_{\mathrm{EI}}(\boldsymbol{x};\zeta) = \begin{cases} (\mu_n(\boldsymbol{x}) - \tau)\Phi\left(\frac{\mu_n(\boldsymbol{x})-\tau}{\sigma_n(\boldsymbol{x})}\right) + \sigma_n(\boldsymbol{x})\phi\left(\frac{\mu_n(\boldsymbol{x})-\tau}{\sigma_n(\boldsymbol{x})}\right) & \sigma_n(\boldsymbol{x}) > 0 \\ 0 & \text{other} \end{cases} \tag{18}$$

where ζ is the sample dataset, $\Phi(\cdot)$ is the cumulative distribution function of standard normal distribution, $\phi(\cdot)$ is the standard normal distribution probability density function, $\mu(\cdot)$ is the posterior mean function, $\sigma_n(\cdot)$ is the posterior variance function, and τ is the current optimal value.

The pseudocode of Bayesian optimization is as follows:

4. Validation of the Effectiveness of a Processing Quality Prediction Method Based on XGBoost Hyperparameter Optimization

The complex geometry of marine diesel engine pistons necessitates numerous processing procedures and sophisticated technologies. Each process's parameters contribute differently to the piston's overall quality. Unlike other diesel engine pistons, marine diesel engine pistons are produced and processed in smaller batches, complicating and destabilizing the process. Currently, there is no systematic method for identifying key processing features, and the control effect on processing quality is inadequate. Therefore, identifying the critical quality characteristics that significantly impact piston processing quality is crucial. Based on this identification, applying advanced quality prediction methods can enhance the stability of the processing process and the reliability of product quality. To clearly illustrate the identification and prediction process of key quality characteristics, this paper introduces a flowchart (Figure 2) to enhance understanding and facilitate the application of these methods. This approach aims to optimize the processing quality of marine diesel engine pistons.

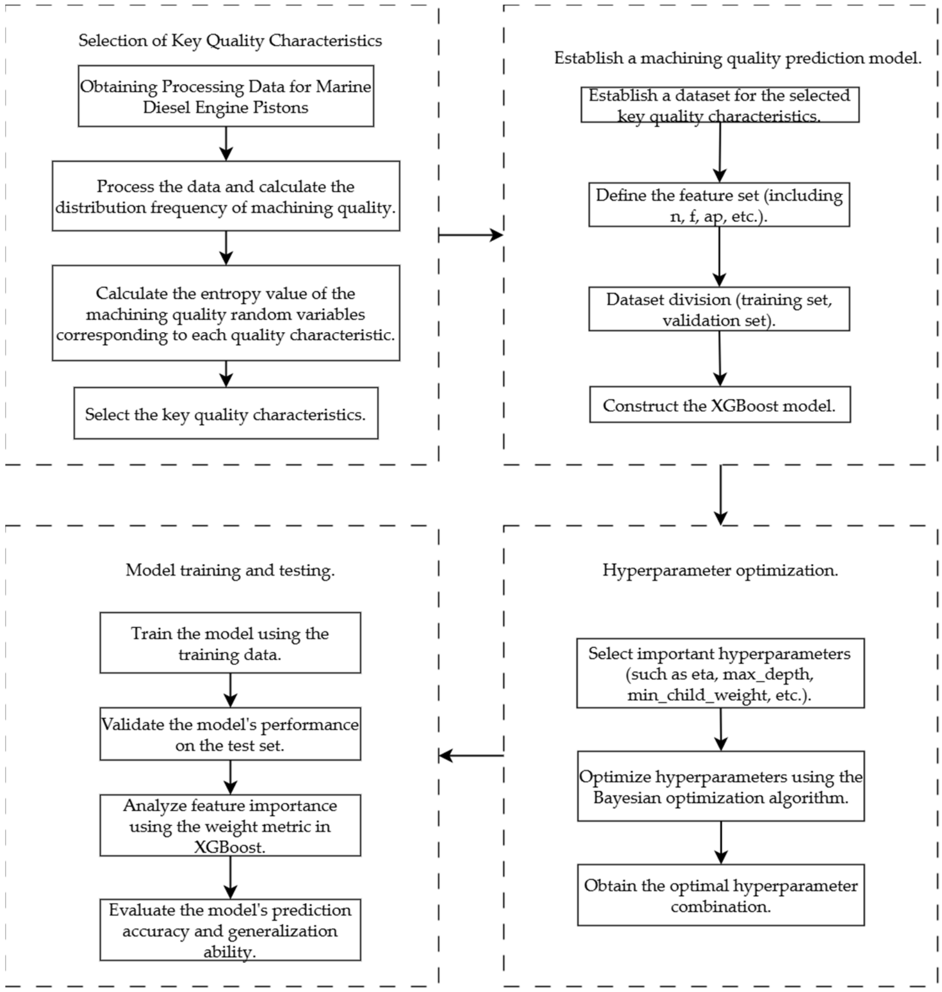

Figure 2. Flowchart of key quality characteristic identification and quality prediction method for marine diesel engine piston machining based on information entropy and XGBoost hyperparameter optimization.

4.1. Identification of Key Processing Characteristics of Marine Diesel Engine Piston

A piston can be divided into three parts: the top, the head, and the skirt. Figure 3 shows the piston's structure. Due to the large size of marine medium-speed diesel engine pistons, they are typically processed in two parts. The piston skirt, being a thin-walled part with poor rigidity and a complex structure, requires a more intricate machining process. For crucial quality characteristics, it is essential to control the size, shape, and surface roughness within tight tolerance ranges to ensure the piston's machining quality and performance.

The arrangement of piston process procedures should follow basic principles: rough machining first, then finishing, and processing the benchmark first, then other features. The machining process of a piston can be broadly divided into three stages: rough machining, semi-finishing, and finishing. The main processes of rough machining include rough turning of the outer circle and end face, rough boring, and milling of both sides of the plane. Semi-finishing processes involve semi-finish turning of the cylindrical end face and ring groove and semi-finish boring. Finishing processes include finishing the end face and stop and fine model line and fine boring. After these primary processes, auxiliary processes such as trimming and deburring, coloring, and flaw detection in the skirt are necessary.

The inner surface of the skirt must be free of defects like cracks and cold scars. Additionally, the skirt undergoes graphitization treatment, and the surface receives anti-rust treatment.

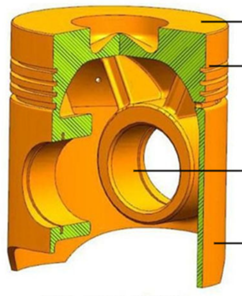

Figure 3. Piston structure.

The comprehensive process card for a specific type of marine medium-speed diesel engine piston is shown in Table 2.

Table 2. Piston comprehensive process card.

Process Number	Description of Operation	Processing Units
1	Forging blank	-
2	Rough turning of the outer diameter and end face and drilling	Lathe
3	Ultrasonic inspection is carried out according to GB/T 6519-2000 Grade A. If oxide films, non-metallic inclusions, pores, and cracks in the macrostructure are non-compliant, an inspection can be performed according to GB/T 6519-2000 Grade AA	-
4	Scribing	Scribing table
5	Rough boring	Digital display boring machine
6	Finish turning of the end face and shoulder	CNC lathe
7	Semi-finish turning of the outer diameter and end face, grooving, counterboring, and reaming	Turn-mill center
8	Drilling and reaming	CNC vertical milling machine
9	Semi-finish boring of holes, milling oil grooves, and arcs	Machining center
10	Milling of two side planes	Digital display boring machine
11	Vibration stress relief	-
12	Finishing the shoulder	Lathe
13	Precision turning of the contour line	CNC lathe
14	Finish boring of holes	Horizontal machining center
15	Milling of two side planes and rounded corners	Digital display boring machine
16	Finishing and deburring	Benchwork finishing
17	Dye penetrant inspection of the skirt interior	-
18	Final inspection	Inspection platform
19	Cleaning and packaging	Cleaning machine
20	Skirt graphite treatment	-

The proposed method was applied to analyze a sample of machining data from 29 marine medium-speed diesel engine pistons processed by a particular company in the same batch. Table 3 shows a portion of the organized data, with the header including process, workstation, dimension type, design dimension, upper deviation, lower deviation, and each part number. Among these, the data for upper deviation and lower deviation significantly impact subsequent calculations.

Table 3. Partially organized piston machining data.

Operation	Operation Step	Quality Characteristic Number	Design Dimension	Upper Deviation	Lower Deviation	Part 1	Part 2	Part 3
2	1	1	80.000	−0.200	−0.300	79.700	79.800	79.700
	2	2	283.000	0.150	−0.150	283.100	283.080	283.100
		3	299.000	0.300	−0.300	299.200	299.100	299.200
5	1	4	105.000	0.100	−0.100	104.960	104.980	104.960
		5	163.500	0.300	−0.300	163.620	163.600	163.620
6	1	6	250.500	0.045	0.000	250.530	250.530	250.540
		7	283.500	0.100	0.000	283.600	283.500	283.600

To analyze the machining quality, divide the length of each machining quality distribution interval corresponding to each quality characteristic in Table 3 into multiple equally spaced sub-intervals. Consider machining quality observations falling into the same sub-interval as belonging to the same category. Each characteristic's machining quality can be viewed as a discrete random variable denoted by uppercase English letters. By examining the distribution of machining quality observations falling into sub-intervals for parts 1 to 29, the distribution frequencies of machining quality in each sub-interval can be determined, as shown in Figure 4. According to the law of large numbers, frequencies converge to probabilities as the sample size increases, implying that frequencies approximate the corresponding probabilities in this context.

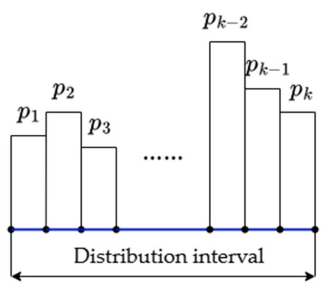

Figure 4. Distribution frequency of quality characteristic observations.

Bringing the approximate machining quality distribution probabilities of each quality characteristic from Table 3 into Equation (1) allows us to calculate the entropy values of the machining quality random variables corresponding to each quality characteristic, as shown in Figure 5.

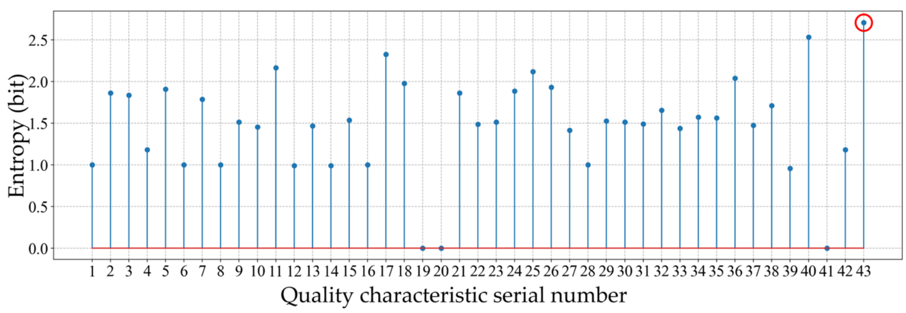

Figure 5. Entropy values of each quality characteristic.

As seen in Figure 5, the entropy value corresponding to the 43rd quality feature in Table 3 is the largest (marked by a red circle). The machining quality of the No. 43 quality feature fluctuates significantly, indicating poor machining consistency. Therefore, we select this quality feature as the critical quality feature.

Critical quality characteristics are not fixed. Once key quality features are identified, we need to optimize their processing parameters. After optimization, the entropy value corresponding to the quality feature decreases, and it may no longer be a critical quality feature. In other words, the screening process for crucial quality characteristics is dynamic and iterative.

4.2. Quality Prediction of Key Machining Characteristics of Marine Diesel Engine Piston

Establishing a processing quality prediction model for key quality characteristics can evaluate the processing quality before machining and reduce the economic losses caused by substandard processing quality in production. Based on information entropy, the critical quality feature identified is the No. 43 quality feature in process 15 (milling the plane and fillet on both sides). The processing equipment used in process 15 is a T611C digital display boring and milling machine, with its basic parameters shown in Table 4.

Table 4. Parameters of T611C digital display boring machine.

Spindle Diameter	**Spindle Motor Power**	**Machining Height**	**Table Load Capacity**
110 mm	6.5–8 Kw	1800 mm	5 t
Three-axis travel	**Boring spindle speed range**	**Table size**	**Facing head tool holder machining diameter**
800 × 1200 × 1100 mm	12–950 rpm	1010 × 1320 mm	630 mm
Forming Lathe Tool R16 (Left)	**Forming Lathe Tool R16 (Right)**	-	-
16 mm	16 mm	-	-

To reduce processing times and economic costs and to obtain a training dataset uniformly distributed in the sample space, an orthogonal experiment with three factors and five levels is used. This results in 125 experimental schemes, from which five sets of non-training data are selected as verification datasets. Cutting speed (n), feed speed (f), and depth of cut (ap) significantly influence the machining process. Su et al. [37] and Yang et al. [38] studied the relationship between these factors and machining error, while Qu et al. [8] analyzed the influence of interactions between processes on machining accuracy. Cutting speed is also an important factor affecting machining quality during the processing of parts, and it can be calculated using the following Formula (19):

$$v = \frac{\pi \cdot n \cdot D}{1000} \tag{19}$$

where v represents the cutting speed in m/min, n is the spindle speed in r/min, and D is the machining diameter of the boring tool in mm.

Therefore, each data point in the dataset should include features such as spindle speed n, feed rate f, and depth of cut a_p, as well as cutting speed v and factors from prior machining operations that influence the current machining process. In Section 4.1, the screening of key quality characteristics of marine medium-speed diesel engine pistons was completed. Based on Table 2 and the piston machining process card, we can abstract the machining process, features, and quality characteristics into vertices in a graph, with the relationships among processes, machining features, and quality features as edges between vertices [39]. Finally, the machining process network of the piston is illustrated in Figure 6.

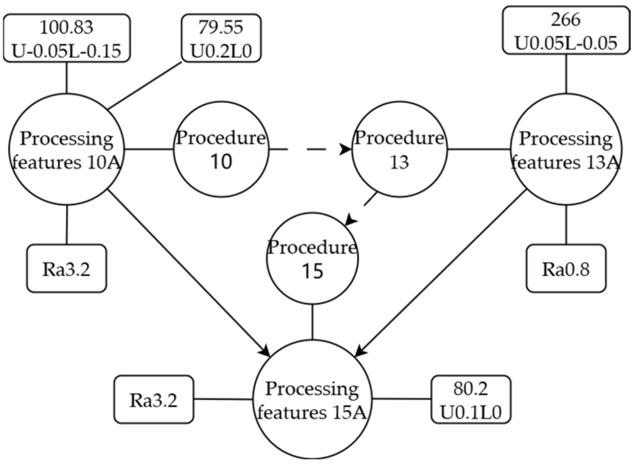

Figure 6. Partial piston machining process network.

As shown in Figure 5, both the preceding machining feature 10A and the machining feature 13A influence the machining feature 15A. The machining feature 10A serves as the positioning reference for the machining feature 15A, while the roughing process of the machining feature 15A affects the machining feature 13A. Consequently, the machining error ε_1 and the roughing error ε_2 of the positioning reference are the other two characteristics of each piece of data. The data are labeled with the machining quality of the critical quality feature, i.e., the actual machining dimension measurement.

Since the XGBoost machine learning algorithm used in this paper is based on the decision tree algorithm, the dataset does not need normalization during preprocessing; it only requires anomaly detection. The box plot method is used to detect abnormal data in the training dataset. Some of the training data after anomaly detection are shown in Table 5.

Table 5. Partial training dataset after anomaly detection.

Serial Number	Input						Output
	n (r/min)	v (m/min)	f (mm/min)	a_p (mm)	ε_1 (mm)	ε_2 (mm)	Processing Quality (mm)
1	500	50.265	20	0.30	−0.035	0.064	80.211
2	500	50.265	25	0.40	−0.032	0.117	80.209
3	500	50.265	30	0.50	−0.030	0.174	80.207
...	...	...	...	...	...	...	...
25	900	90.478	40	0.5	−0.024	0.103	80.258

Before training the model, it is necessary to identify which hyperparameters should be optimized. XGBoost has many hyperparameters, and some of the more critical ones are listed in Table 6. When using Bayesian optimization to optimize the hyperparameters of the XGBoost model, the more hyperparameters that are optimized, the greater the computing power required. Therefore, optimizing essential hyperparameters can significantly improve model performance with less computation. This paper focuses on optimizing the first five hyperparameters—eta, min_child_weight, max_depth, alpha, and lambda—as shown in Table 6.

Table 6. XGBoost partial hyperparameters.

Parameter Name	Default Value	Parameter Description
eta	0.3	Learning rate, eta $\in [0,1]$, commonly used values are 0.01~0.2.
min_child_weight	1	Child nodes contain the smallest sum of instance weights, min_child_weight $\in [0,\infty]$.
max_depth	6	Maximum depth of tree, max_depth $\in [1,\infty]$, commonly used values are 3~10.
α	1	The L1 regular term of the weight, the larger the value, the more conservative the model is.
λ	1	The L2 regular term of the weight, the larger the value, the more conservative the model is.
γ	0	Minimum loss reduction required for leaf node splitting, gamma $\in [0,\infty]$.
max_delta_step	0	Maximum incremental step size allowed for each tree weight estimation, max_delta_step $\in [0,\infty]$.
scale_pos_weight	1	The balance of positive and negative sample weights is controlled when the categories are unbalanced.
subsmaple	1	Sampling rate of training samples, subsmaple $\in (0,1]$.
......	......	......

The dataset in Table 5 is divided, and the model's hyperparameters are optimized with the minimization of the evaluation results of five-fold cross-validation as the optimization goal. During the hyperparameter optimization process, each parameter's setting is shown in Table 7.

Table 7. Parameter settings of Bayesian optimization.

Initial Random Sampling Count	Maximum Number of Iterations	Selected Acquisition Function	Eta Optimization Interval
10	50	EI	(0, 1]
min_child_weight optimization interval	**max_depth optimization interval**	**α optimization interval**	**λ optimization interval**
(0, 10)	(3, 10)	(1, 5)	(1, 5)

Table 8 records the data of the first five iterations of Bayesian optimization and the obtained optimal hyperparameters (last row). The objective function values corresponding to different hyperparameter combinations vary, and the hyperparameter set corresponding to the smallest objective function value is considered optimal. It should be noted that the objective function value is the negative of the five-fold cross-validation test results. This transformation is necessary because the optimization program aims to maximize the objective function, so taking its negative converts it into the minimization of the five-fold cross-validation test results.

Table 8. Data of the first five iterations of Bayesian optimization and optimal solutions of hyperparameters.

Objective Function Value	Eta	Max_Depth	Min_Child_Weight	α	λ
−0.031	1.000	8.206	7.092	1.646	3.499
−0.039	1.000	9.282	5.922	3.498	1.988
−0.037	0.038	9.092	4.810	1.000	3.789
−0.030	0.635	8.231	6.586	4.916	2.985
−0.033	1.000	10.000	4.326	3.799	3.000
−0.013	0.447	5.710	4.894	2.457	1.288

After obtaining the optimal hyperparameters, the XGBoost model can be trained, with the training process illustrated in Figure 7.

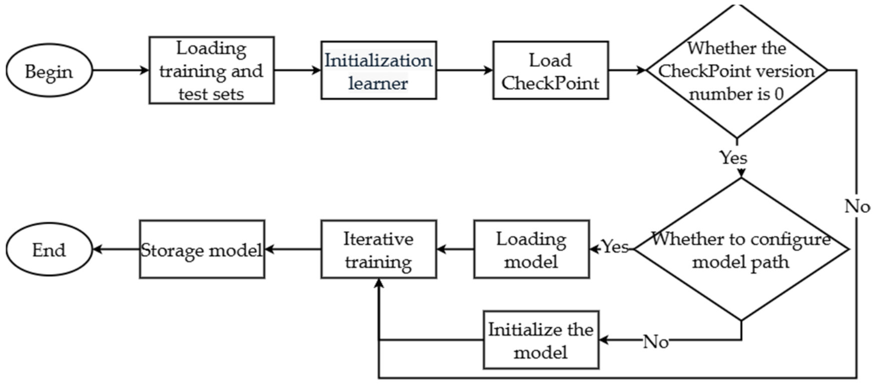

Figure 7. XGBoost model training process.

The XGBoost model needs to be verified on the training set, and its prediction accuracy and generalization ability on datasets other than the training set must be tested. The test set data is shown in Table 9. For testing the XGBoost model and finding the optimal parameters, it is necessary to predefine the testing range. Therefore, we set five groups of variables for parameter optimization, as shown in Table 9.

Table 9. Test dataset.

Serial Number	n (r/min)	v (m/min)	f (mm/min)	a_p (mm)	ε_1 (mm)	ε_2 (mm)	Processing Quality (mm)
1	500	50.265	25	0.30	0.117	−0.033	80.211
2	600	60.319	30	0.40	0.099	−0.017	80.217
3	700	70.372	40	0.50	0.023	−0.027	80.238
4	800	80.425	20	0.45	0.126	−0.030	80.200
5	900	90.478	35	0.35	0.019	−0.044	80.283

The XGBoost model is trained iteratively 50 times. As the number of iterations increases, the change in root mean square error (RMSE) of the model on the training set and test set is shown in Figure 8. It can be seen from the figure that, with an increasing number of training iterations, the RMSE of the model on both the training set and the test set shows a decreasing trend. In the first 30 iterations, the RMSE of the model on both sets decreases rapidly. With the same number of iterations, the RMSE of the model on the training set is smaller than that on the test set.

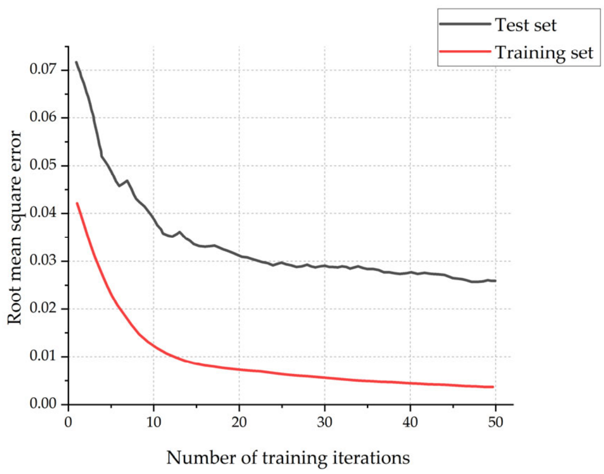

Figure 8. Change of root mean square error during model training.

The predicted and actual values of the trained XGBoost model on the test set are shown in Figure 9. The predicted values are consistent with the trends of the actual values, and the numerical differences are slight. The RMSE of the XGBoost model on the test set is 0.0307, which meets the accuracy requirements.

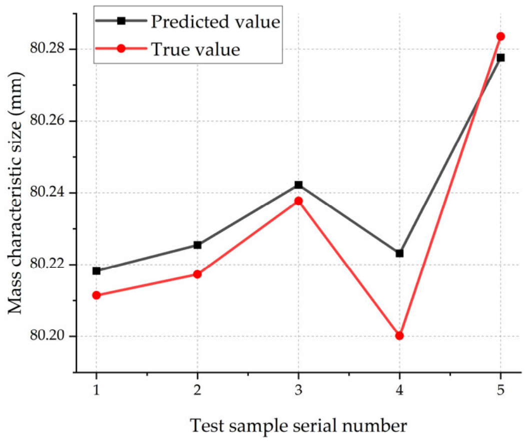

Figure 9. Predicted and true values of the model on the test set.

XGBoost provides five indicators to measure the importance of features: the number of times a feature is used as a split feature (weight), the average feature return (gain), the average feature usage coverage (cover), the total feature return (total_gain), and the total feature usage coverage (total_cover). In this paper, weight is used as the evaluation index of feature importance, and each feature is sorted according to its importance in the XGBoost model. The ranking results are shown in Figure 10.

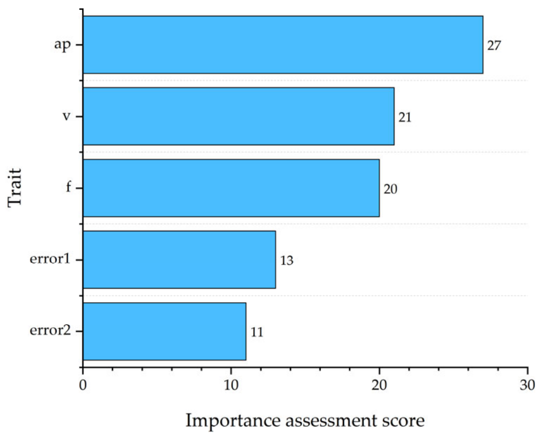

Figure 10. Ranking of importance of model features.

Default values are assigned to each hyperparameter in the XGBoost framework, as shown in Table 6. To verify the effectiveness of hyperparameter optimization based on Bayesian optimization, the model is re-trained (referred to as Model 2) using default hyperparameters and compared with the previously trained model (referred to as Model 1). The root mean square error (RMSE) of Model 2 on the test set is 0.042, which is 0.03 higher than that of Model 1 (0.012). The prediction results of Models 1 and 2 on the test set are shown in Figure 11. From both the figure and the RMSE values, it is evident that Model 1, which uses Bayesian optimization, outperforms Model 2 in terms of generalization ability and prediction accuracy. This demonstrates the effectiveness of Bayesian optimization.

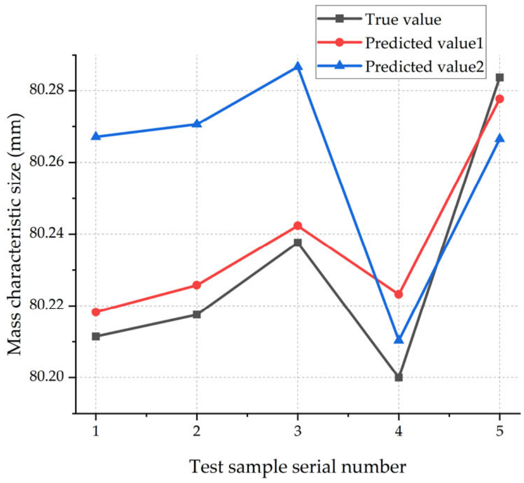

Figure 11. Prediction comparison of Models 1 and 2 on the test set.

5. Results and Discussion

In this study, we use the machining of a marine diesel engine piston as an example to demonstrate our proposed method for identifying key machining features and constructing a machining quality prediction model. Through the information entropy method, we identified the 43rd quality feature as the key quality feature. Subsequently, we employed the XGBoost algorithm to establish the corresponding machining quality prediction model. The optimal hyperparameters for this model were determined using Bayesian optimization and validated on a test set, resulting in relatively accurate prediction outcomes.

The analysis indicates that the 43rd quality feature has the highest entropy value, suggesting significant quality fluctuations during the machining process, which establishes it as the key quality feature. These large fluctuations in the 43rd quality feature are associated with various factors, including machining processes, equipment stability, and process parameter settings. By further optimizing these factors, we expect to enhance the consistency of machining quality.

Through training the XGBoost model, we observed a decreasing trend in the root mean square error (RMSE) on both the training and test sets. During the first 30 iterations, the RMSE decreased rapidly for both sets, with the RMSE on the training set consistently lower than that on the test set. After 50 iterations, the model's RMSE reached 0.0307, meeting the required accuracy threshold. The observed decreasing trend in RMSE across different iterations indicates that XGBoost is highly adaptable in addressing machining quality prediction problems. Notably, the model optimized through Bayesian optimization outperformed the model with default parameter settings on the test set, underscoring the significance of hyperparameter optimization in enhancing model accuracy.

An analysis of the feature importance indicators in the XGBoost model revealed that the feature ranking met our expectations, further validating the model's reliability. Key process parameters, including spindle speed, feed rate, and cutting depth, significantly impacted machining quality. Adjustments to these parameters during optimization directly influenced the model's prediction accuracy, reinforcing their crucial role in machining quality.

Future research should aim to expand the scale of the training dataset to enhance the model's generalization capabilities. Additionally, factors such as ambient temperature and tool wear, which may also affect machining quality, should be considered to improve the model's applicability. Furthermore, integrating other machine learning algorithms, such as deep learning models, for multi-model comparisons could lead to more effective machining quality prediction methods.

6. Conclusions

This study examines key quality characteristics during the machining process of mechanical products and predicts their processing quality. We propose a method for identifying these key machining quality characteristics and predicting their quality using information entropy and hyperparameter optimization of the XGBoost algorithm. The XGBoost model demonstrates effective prediction of the machining quality of marine diesel engine pistons while also identifying essential quality characteristics. The results and discussions indicate that optimizing both process parameters and model hyperparameters can significantly enhance the consistency of processing quality and the predictive accuracy of the model. This research provides a scientific basis for further optimizing the machining process of marine diesel engine pistons. The main content of this study is outlined as follows:

1. Key machining quality characteristics are identified using information entropy, effectively addressing the shortcomings of other commonly used analytical methods that are heavily influenced by subjective factors from decision makers.
2. In constructing the quality prediction model, this study integrates the XGBoost algorithm with Bayesian hyperparameter optimization for the first time, achieving a balance between prediction efficiency and accuracy of results.
3. Using the machining process of a marine diesel engine piston as an example, the model is evaluated using the root mean square error (RMSE). The results indicate that the RMSE of the model on the training dataset is 0.012, while on the testing dataset, it is 0.0307, demonstrating high predictive accuracy. Experimental results further validate the excellent performance and strong generalization ability of the proposed prediction model.

The research results present an effective quality prediction model specifically designed for multi-process machining. This model addresses the shortcomings of traditional quality prediction methods, such as their complex calculations and limited applicability. It is particularly beneficial for analyzing and predicting the machining of parts with intricate quality characteristics, which are often challenging for traditional models to handle. Moreover, this model not only facilitates the analysis and prediction of machining quality but also serves as a tool for optimizing machining processes in the future, thereby enhancing quality control in intelligent manufacturing. Furthermore, this method has shown significant potential to enhance machining quality while also reducing rework and scrap rates in the quality control of ship piston machining. This improvement is crucial for boosting the overall competitiveness of the manufacturing industry.

Author Contributions: Conceptualization, C.G. and D.Q.; methodology, D.Q.; software, C.G.; validation, Y.Z. (Yong Zhan) and D.Q.; formal analysis, Y.Z. (Yong Zhan), H.Z., Y.Z. (Yuting Zhang) and W.L.; investigation, C.G.; resources, D.Q.; data curation, C.G.; writing—original draft preparation, C.G.; writing—review and editing, D.Q., Y.Z. (Yong Zhan), Y.Z. (Yuting Zhang), W.L. and H.Z.; visualization, C.G.; supervision, Y.Z. (Yong Zhan); project administration, D.Q.; funding acquisition, D.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Designing and Verifying Key Technologies of Marine Engines High Reliability in Major Special Project Ship Research Program of the Ministry of Industry and Information Technology (MIIT) of the People's Republic of China under Grant CBG5N21-2-1.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available on request from the corresponding author, Zhan, upon reasonable request.

Conflicts of Interest: We have no competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Zhang, Y.; Huang, Y.; Shao, W.; Ming, W. Research on MCVE piston machining and process parameter optimization. *Int. J. Adv. Manuf. Technol.* **2017**, *93*, 3955–3966. [CrossRef]
2. Peniwati, K. Group decision making: Drawing out and reconciling differences. *Int. J. Anal. Hierarchy Process* **2017**, *9*, 385–389. [CrossRef]
3. Deng, Y.; Chen, J.; Liu, X.; Zhang, Q.; Chen, S.; Wu, L. Study of Deformation Factors Extracting on Flexible Material Machining by Analytical Hierarchy Process. *J. Mech. Eng.* **2016**, *52*, 161–169. [CrossRef]
4. Zhang, J. Fuzzy analytic hierarchy process. *Fuzzy Syst. Math.* **2000**, *14*, 80–88.
5. Russell, R.S.; Taylor, B.W. *Operations and Supply Chain Management*; John Wiley & Sons: Hoboken, NJ, USA, 2019; pp. 97–99.
6. Zhang, H. Research on Regulation Mechanism Based on Multiple Quality Characteristics and Manufacturing Process of Conical Pick. Ph.D. Thesis, Taiyuan University of Science and Technology, Taiyuan, China, 2021.
7. Xing, X. Physical entropy, information entropy and their evolution equations. *Sci. China Ser. A Math.* **2001**, *44*, 1331–1339. [CrossRef]
8. Qu, D.; Ma, J.; Wang, Y.; Zhan, Y. Reliability analysis of multi-process machining based on information entropy. *Appl. Sci.* **2023**, *13*, 3133. [CrossRef]
9. Wang, M.; Bian, Y. Evaluation Method of Stationarity for NC Machining Process Based on Information Entropy. In Proceedings of the 2020 3rd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM), Shanghai, China, 4–6 December 2020; IEEE: Piscataway, NJ, USA; pp. 287–291. [CrossRef]
10. Zhang, Y.; Guo, G.; Liu, J. Fault Root Cause Tracking of the Mechanical Components of CNC Lathes Based on Information Transmission. *Sensors* **2023**, *23*, 4418. [CrossRef]
11. Zajac, J.; Duplak, J.; Duplakova, D.; Cizmar, P.; Olexa, I.; Bittner, A. Prediction of cutting material durability by T = f (vc) dependence for turning processes. *Processes* **2020**, *8*, 789. [CrossRef]
12. Makhfi, S.; Dorbane, A.; Harrou, F.; Sun, Y. Prediction of cutting forces in hard turning process using machine learning methods: A case study. *J. Mater. Eng. Perform.* **2023**, 1–17. [CrossRef]
13. Alajmi, M.S.; Almeshal, A.M. Prediction and optimization of surface roughness in a turning process using the ANFIS-QPSO method. *Materials* **2020**, *13*, 2986. [CrossRef]
14. Han, Z.; Zhao, J.; Leung, H.; Ma, K.F.; Wang, W. A review of deep learning models for time series prediction. *IEEE Sens. J.* **2019**, *21*, 7833–7848. [CrossRef]
15. Janiesch, C.; Zschech, P.; Heinrich, K. Machine learning and deep learning. *Electron. Mark.* **2021**, *31*, 685–695. [CrossRef]
16. Yang, H.; Pan, Z.; Bai, W. Review of Time Series Prediction Methods. *Comput. Sci.* **2019**, *46*, 22–28.
17. Chicco, D.; Warrens, M.J.; Jurman, G. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Comput. Sci.* **2021**, *7*, e623. [CrossRef]
18. Choi, R.Y.; Coyner, A.S.; Kalpathy-Cramer, J.; Chiang, M.F.; Campbell, J.P. Introduction to machine learning, neural networks, and deep learning. *Transl. Vis. Sci. Technol.* **2020**, *9*, 14. [CrossRef] [PubMed]
19. Rokach, L. Decision forest: Twenty years of research. *Inf. Fusion* **2016**, *27*, 111–125. [CrossRef]
20. Biau, G.; Scornet, E. A random forest guided tour. *Test* **2016**, *25*, 197–227. [CrossRef]
21. Wang, H.; Liu, H.; Zhang, X. Development trend of support vector machine and applications on the field of computer science. In Proceedings of the 2016 International Conference on Computer Engineering, Information Science & Application Technology (ICCIA 2016), Guilin, China, 24–25 September 2016; Volume 56, pp. 46–55.
22. Turgut, Y.; Erdem, M. Forecasting of retail produce sales based on XGBoost algorithm. In Proceedings of the Industrial Engineering in the Internet-of-Things World: Selected Papers from the Virtual Global Joint Conference on Industrial Engineering and Its Application Areas, GJCIE 2020, Virtual, 14–15 August 2020; pp. 27–43.
23. Ahmed, F.; Kim, K.Y. Data-driven weld nugget width prediction with decision tree algorithm. *Procedia Manuf.* **2017**, *10*, 1009–1019. [CrossRef]
24. Ye, X.; Wu, X.; Guo, Y. Real-time quality prediction of casting billet based on random forest algorithm. In Proceedings of the 2018 IEEE International Conference on Progress in Informatics and Computing (PIC), Suzhou, China, 14–16 December 2018; pp. 140–143.
25. Yu, H.; Qin, S.; Ding, G.; Jiang, L.; Qin, H. Research on Prediction Technology for Tool Rotation ProfileErrors in Flank Milling. *China Mech. Eng.* **2020**, *31*, 306–313.
26. Huang, Y.P.; Yen, M.F. A new perspective of performance comparison among machine learning algorithms for financial distress prediction. *Appl. Soft Comput.* **2019**, *83*, 105663. [CrossRef]
27. Liao, X.; Cao, N.; Li, M.; Kang, X. Research on short-term load forecasting using XGBoost based on similar days. In Proceedings of the 2019 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS), Changsha, China, 12–13 January 2019; pp. 675–678.
28. Demir, S.; Sahin, E.K. Predicting occurrence of liquefaction-induced lateral spreading using gradient boosting algorithms integrated with particle swarm optimization: PSO-XGBoost, PSO-LightGBM, and PSO-CatBoost. *Acta Geotech.* **2023**, *18*, 3403–3419. [CrossRef]
29. Truong, G.T.; Choi, K.K.; Nguyen, T.H.; Kim, C.S. Prediction of shear strength of RC deep beams using XGBoost regression with Bayesian optimization. *Eur. J. Environ. Civ. Eng.* **2023**, *27*, 4046–4066. [CrossRef]

30. Stuke, A.; Rinke, P.; Todorović, M. Efficient hyperparameter tuning for kernel ridge regression with Bayesian optimization. *Mach. Learn. Sci. Technol.* **2021**, *2*, 035022. [CrossRef]
31. Xiong, X.; Guo, X.; Zeng, P.; Zou, R.; Wang, X. A short-term wind power forecast method via xgboost hyper-parameters optimization. *Front. Energy Res.* **2022**, *10*, 905155. [CrossRef]
32. Cover, T.M. *Elements of Information Theory*; John Wiley & Sons: Hoboken, NJ, USA, 1999.
33. Joy, T.T.; Rana, S.; Gupta, S.; Venkatesh, S. Fast hyperparameter tuning using Bayesian optimization with directional derivatives. *Knowl. Based Syst.* **2020**, *205*, 106247. [CrossRef]
34. Perdikaris, P.; Karniadakis, G.E. Model inversion via multi-fidelity Bayesian optimization: A new paradigm for parameter estimation in haemodynamics, and beyond. *J. R. Soc. Interface* **2016**, *13*, 20151107. [CrossRef]
35. Cui, J.; Yang, B. Survey on Bayesian Optimization Methodology and Applications. *J. Softw.* **2018**, *29*, 3068–3090.
36. Li, Y.; Zhang, Y.; Wang, J. Survey on Bayesian Optimization Methods for Hyper-parameter Tuning. *Comput. Sci.* **2022**, *49*, 86–92. [CrossRef]
37. Su, X.; Guo, R.; Li, Y.; Bai, L.; Lai, F.; Xu, J. Processing Quality Prediction of Precision Milling Based on XGBoost. *Mach. Build. Autom.* **2023**, *52*, 72–76.
38. Yang, W. Research on Prediction and Control Method of Machining Quality of Marine Diesel Engine Body. Master's Thesis, Jiangsu University of Science and Technology, Zhenjiang, China, 2022.
39. Wang, L.; Dai, W.; Sun, D.; Zhao, Y. Risk evaluation for a manufacturing process based on a directed weighted network. *Entropy* **2020**, *22*, 699. [CrossRef]