

Prediction of surface roughness using deep learning and data augmentation

Prediction of
surface
roughness

Miaoxian Guo, Shouheng Wei, Chentong Han and Wanliang Xia
University of Shanghai for Science and Technology, Shanghai, China, and
Chao Luo and Zhijian Lin
Aplos Machines Manufacturing (Shanghai) Co. Ltd, Shanghai, China

Received 24 October 2023
Revised 23 December 2023
Accepted 2 January 2024

Abstract

Purpose – Surface roughness has a serious impact on the fatigue strength, wear resistance and life of mechanical products. Realizing the evolution of surface quality through theoretical modeling takes a lot of effort. To predict the surface roughness of milling processing, this paper aims to construct a neural network based on deep learning and data augmentation.

Design/methodology/approach – This study proposes a method consisting of three steps. Firstly, the machine tool multisource data acquisition platform is established, which combines sensor monitoring with machine tool communication to collect processing signals. Secondly, the feature parameters are extracted to reduce the interference and improve the model generalization ability. Thirdly, for different expectations, the parameters of the deep belief network (DBN) model are optimized by the tent-SSA algorithm to achieve more accurate roughness classification and regression prediction.

Findings – The adaptive synthetic sampling (ADASYN) algorithm can improve the classification prediction accuracy of DBN from 80.67% to 94.23%. After the DBN parameters were optimized by Tent-SSA, the roughness prediction accuracy was significantly improved. For the classification model, the prediction accuracy is improved by 5.77% based on ADASYN optimization. For regression models, different objective functions can be set according to production requirements, such as root-mean-square error (RMSE) or MaxAE, and the error is reduced by more than 40% compared to the original model.

Originality/value – A roughness prediction model based on multiple monitoring signals is proposed, which reduces the dependence on the acquisition of environmental variables and enhances the model's applicability. Furthermore, with the ADASYN algorithm, the Tent-SSA intelligent optimization algorithm is introduced to optimize the hyperparameters of the DBN model and improve the optimization performance.

Keywords Multi-sensor fusion, Surface quality, Digital signal processing, Feature engineering, Neural network, Parameter optimization

Paper type Research paper

1. Introduction

The machinery manufacturing industry is undergoing a digital and intelligent revolution as the Internet and artificial intelligence become more prevalent. Traditional computer numerical control (CNC) machining is transforming into intelligent CNC machining. The fundamental premise of intelligent CNC machining is to produce an efficient, completely automatic system, which can be tackled in two steps: the first is infrastructure monitoring, which involves real-time tracking of the CNC system's core components, and the second is quality diagnostics utilizing machine learning (ML) technology.



© Miaoxian Guo, Shouheng Wei, Chentong Han, Wanliang Xia, Chao Luo and Zhijian Lin. Published in *Journal of Intelligent Manufacturing and Special Equipment*. Published by Emerald Publishing Limited. This article is published under the Creative Commons Attribution (CC BY 4.0) licence. Anyone may reproduce, distribute, translate and create derivative works of this article (for both commercial and non-commercial purposes), subject to full attribution to the original publication and authors. The full terms of this licence may be seen at <http://creativecommons.org/licences/by/4.0/legalcode>

Simultaneously, the quality standards of mechanical products are increasing, particularly in high-end complicated areas such as aerospace, precision manufacturing and precision medical equipment. In rare circumstances, the quality of mechanical items has a significant impact on product service life. Surface roughness has a significant impact on the fatigue strength, wear resistance and surface hardness of mechanical products (He *et al.*, 2015), hence it is frequently used to assess product quality. Surface roughness is caused by a complex mechanism dependent on specific manufacturing procedures, and the analytical expression cannot be simply determined (Benardos and Vosniakos, 2002). As a result, many academics are dedicated to the monitoring and prediction of workpiece roughness during the machining process, and numerous technologies are employed to establish the relationship between physical phenomena and surface roughness during the cutting process.

Surface roughness estimation is classified into two categories: classic theory-based modeling and machine-learning-based prediction. The theoretical modeling method establishes a statistical regression or classification model of surface roughness through response surface analysis of experimental data by analyzing the relationship between the surface roughness generation mechanism and process parameters. The ML method achieves real-time roughness prediction by building a one-to-one relationship between data and surface roughness.

Many researchers have done pertinent research in the area of predicting roughness with ML. Agrawal *et al.* (2015) established a surface roughness prediction model by using turning parameters, random forest and multiple regression methods. Abu-Mahfouz *et al.* (2017) used linear and polynomial kernels to introduce fast Fourier transform (FFT) and continuous wavelet transform (CWT) features, as well as statistical features, into support vector machine (SVM) to predict surface roughness classification and compare with K-nearest neighbor, decision tree and random forest classifiers. Zhou *et al.* (2019) predicted surface roughness using a Gradient-boosting regression tree (GBRT) to fit the relationship between process parameters and surface roughness. Lu *et al.* (2019) optimized the internal parameters of the SVM model using the artificial bee colony algorithm (ABC) and matched the surface roughness prediction model based on the SVM model with machining process parameters and tool tip radius.

The neural network can improve the roughness prediction because of its outstanding performance in high dimensional data feature learning efficiency and accuracy. Chen *et al.* (2017) proposed a nested artificial neural network to predict the surface roughness of the turning process. Lin *et al.* (2019) processed vibration signals based on three methods: FFT deep neural network (FFT-DNN), FFT long short-term memory (FFT-LSTM) network and one-dimensional convolutional neural network (1-D CNN), and established three prediction models of surface roughness respectively. Based on the idea of deep learning, Pan *et al.* (2022) discretizes surface roughness and transforms the fitting problem into a classification problem and the relationship between vibration signal and surface roughness is established. Kong *et al.* (2020) found that Standard sparse Bayesian linear regression (SBLR) had the best predictive performance among the four Bayesian linear regression (BLR) models when dimensionality reduction was based on integrated radial basis function based kernel principal component analysis (KPCA_IRBF). Guo *et al.* (2021) propose a hybrid feature selection method that selects features based on their correlation to surface roughness, as well as hardware and time costs. Lv *et al.* (2021) propose an end-to-end deep learning prediction model using a sequential deep learning framework and a LSTM network. Cooper *et al.* (2023) developed a conditional generative adversarial network (CGAN) to synthesize power signals associated with different combinations of process parameters, and the synthesized signal is then used to enhance the measurement signal and develop CNNs to predict machined surface roughness.

Although the previous research has obtained predictions of surface roughness in some specific cases, it is difficult to accurately reflect the surface roughness for the limited sample

data set, and the neural network model needs to be optimized. To improve the accuracy and reliability of surface roughness prediction, a neural network prediction method based on multisensor fusion and intelligent optimization was proposed. By applying deep learning and data enhancement, raw data is collated and trained to produce models that are more applicable and more accurate than the previous ML models.

2. Methodology

In this study, applying variational modal decomposition (VMD), the real-time signals were first processed to extract relevant features of the workpiece surface roughness information. Then, the adaptive synthetic sampling (ADASYN) oversampling algorithm is applied, to balance the data distribution and improve the performance of the prediction model. Deep belief network (DBN) is selected as the prediction model, which consists of multiple layers of interconnected neurons, and the sparrow search method based on the Tent chaotic mapping algorithm is used to train the DBN model. This optimization technique aims to find the optimal weights and biases of the DBN model to improve its predictive ability.

2.1 Feature extraction

To provide candidate features with enough feature selection information to build an accurate workpiece surface roughness monitoring model, this paper extracted 17 time-domain features, 5 frequency-domain features, and 4 time-frequency-domain features from milling force signals and vibration signals in each direction (Guo *et al.*, 2023), resulting in a total of $26 \times 6 = 156$ features (Table 1). There are four other MATLAB properties of spectral kurtosis in addition to these: the mean of the spectral kurtosis, the standard deviation of the spectral kurtosis, the skewness of the spectral kurtosis and the magnitude of the spectral kurtosis.

2.2 Signal processing

By changing the constrained problem into an unconstrained problem, the VMD model (Dragomiretskiy and Zosso, 2014) obtains the optimal solution of the model, including the finite bandwidth of each center frequency and modality. The VMD approach is used to decompose the cut signal, and the energy entropy of each modal component resulting from the decomposition can quantify the change in roughness at different scales.

For the one-dimensional VMD decomposition of M IMFs ($i = 1, 2, \dots, M$) of one-dimensional VMD decompositions, the respective energy and full intensity can be found according to the following equation.

Energy: denotes the magnitude of the signal strength of these IMFs, which is measured by the variance of the data ($\text{Var}(\cdot)$) to measure it:

$$E_i = \text{Var}(\text{IMF}_i) \quad (1)$$

Entropy: indicates the signal complexity of the intrinsic mode function (IMF), measured by the information content of the signal ($H(\cdot)$) to measure:

$$S_i = H(\text{IMF}_i) \quad (2)$$

where the signal entropy is calculated as:

$$H(S) = - \sum_{i=1}^N P(S_i) \log_2(P(S_i)) \quad (3)$$

where $P(S_i)$ denotes the probability of the i th signal in the signal S .

Index	Feature name	Calculation formula
1	Maximum value	x_{max}
2	Minimum value	x_{min}
3	Mean value	$x_{mean} = \frac{1}{N} \sum_{i=1}^N x_i$
4	Median value	x_{med}
5	Peak-to-peak	$x_{p-p} = x_{max} - x_{min}$
6	Average absolute value	$x_{av} = \frac{1}{N} \sum_{i=1}^N x_i $
7	Variance	$x_{var} = \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2$
8	Standard deviation	$x_{std} = \left(\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2 \right)^{1/2}$
9	Root mean square	$x_{rms} = \left(\frac{1}{N} \sum_{i=1}^N x_i^2 \right)^{1/2}$
10	Mean square value	$x_{ms} = \frac{1}{N} \sum_{i=1}^N x_i^2$
11	Root mean square amplitude	$x_{rmsa} = \left(\frac{1}{N} \sum_{i=1}^N x_i ^{1/2} \right)^2$
12	Skewness	$x_{sk} = \frac{1}{n-1} \frac{\sum_{i=1}^N (x_i - \bar{x})^3}{x_{std}^3}$
13	Kurtosis	$x_{ku} = \frac{1}{n-1} \frac{\sum_{i=1}^N (x_i - \bar{x})^4}{x_{std}^4}$
14	Form factor	$S_f = \frac{x_{rms}}{\frac{1}{N} \sum_{i=1}^N x_i }$
15	Peak factor	$C = \frac{x_{p-p}}{x_{rms}}$
16	Impulse factor	$S = \frac{x_{p-p}}{\frac{1}{N} \sum_{i=1}^N x_i }$
17	Clearance factor	$L = \frac{x_{p-p}}{\left(\frac{1}{N} \sum_{i=1}^N x_i ^{1/2} \right)^2}$
18	Center of gravity frequency	$f_c = \frac{\sum_{i=1}^N f_i p_i}{\sum_{i=1}^N p_i}$
19	Frequency variance	$v_f = \frac{\sum_{i=1}^N (f_i - f_c)^2 p_i}{\sum_{i=1}^N p_i}$
20	Frequency standard deviation	$v_s = \sqrt{v_f}$
21	Mean square frequency	$msf = \frac{\sum_{i=1}^N f_i^2 p_i}{\sum_{i=1}^N p_i}$
22	Root mean square frequency	$rmsf = \sqrt{msf}$

Table 1.
List of signal features

Source(s): Authors' own work

The sum of the energy of each component will change dramatically depending on the value of the parameter K. In this study, the best K value of the VMD is determined by comparing the total energy difference.

Because the K value is too small to completely decompose a series of components with orthogonal relationships in the actual machining process, the modal number K value is set to

be increased from 4 to 10, and the energy sum of different K values is shown in Table 2. The milling force signal has the greatest difference in energy values at $K = 8$, and the vibration signal has the greatest difference at $K = 6$, so the modal number of the milling force signal is 8 and the modal number of the vibration signal is 6.

For the X-direction milling force signal original data, as shown in the first row of data in Figure 1, the VMD is carried out with $K = 8$ and $\alpha = 3,600$, and the IMF component time domain and frequency domain plots are shown in Figure 1.

For the X-direction vibration signal original data, the VMD is carried out with $K = 6$ and $\alpha = 3,600$, and the IMF component time domain and frequency domain diagrams are shown in Figure 2.

2.3 Expansion of dataset

The data set obtained from the experiment must be enlarged for classification prediction, as the low number of samples may not be sufficient to meet the training needs of the deeper network. The synthetic minority over-sampling technique (SMOTE) algorithm is used to

Preset resolution	Vibration signal	Milling force signal
4	0.032	0.004
5	0.031	0.011
6	0.605	0.015
7	0.048	0.021
8	0.027	0.094
9	0.041	0.035
10	0.025	0.032

Table 2.
The sum of the energies at different values of K

Source(s): Authors' own work

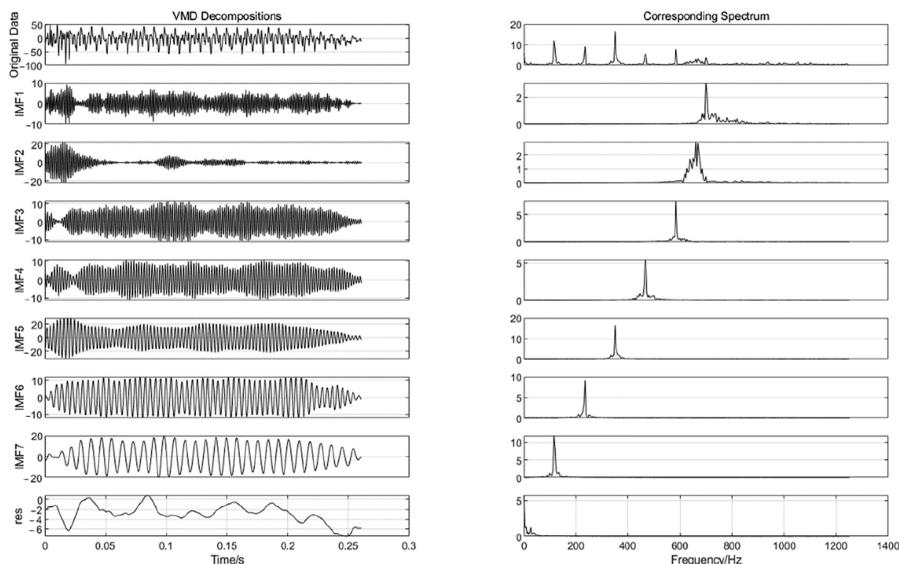


Figure 1.
Time domain diagram and frequency domain diagram of IMF component of VMD

Note(s): Milling force signal in X direction

Source(s): Authors' own work

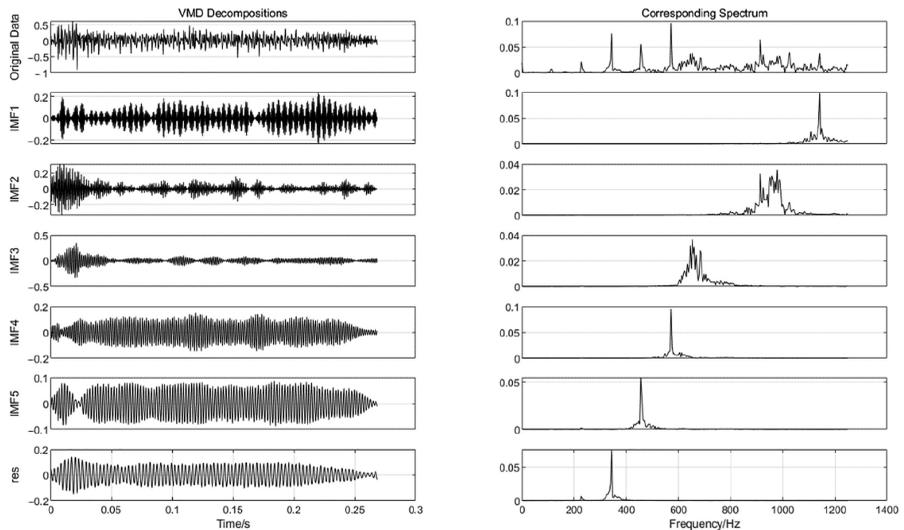


Figure 2.
Time domain and
frequency domain of
IMF component

Note(s): Vibration signal in X direction

Source(s): Authors' own work

generate new samples by intentionally adding additional data points to the dataset depending on the distribution of the original samples (Fernandez *et al.*, 2018), which increases the number of resettable samples to some extent.

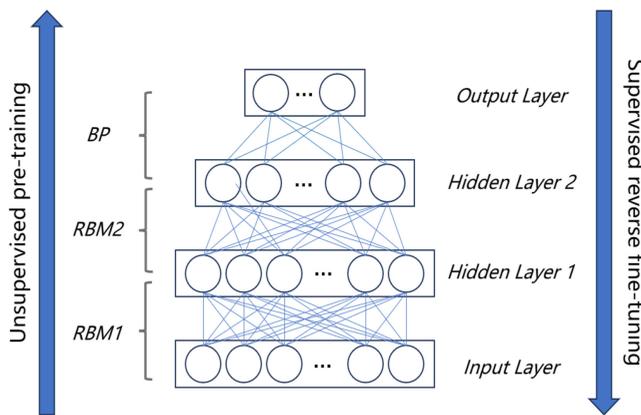
The ADASYN method analyzes the density distribution of a small number of sample categories to generate numerous new composite samples (Ahmed *et al.*, 2022), intending to balance the data. The ADASYN algorithm's specific steps are as follows: (1) Determine the distance between samples; (2) Determine the density of neighboring samples for each sample; (3) Create new synthetic samples and (4) Include the synthetic sample in the original data set.

2.4 Deep Belief Network

The DBN (Hinton and Salakhutdinov, 2006) trains and repeatedly updates the weighting parameters between neurons so that the model corresponds to the maximum probability of the DBN network (Scarpiniti *et al.*, 2021). Stacking the two results in the network, DBN's architecture is influenced by deep structures in neuroscience, to simulate how the human brain processes information.

DBN may gradually learn several levels of abstract feature representation of data by stacking successive restricted Boltzmann machine (RBM) layers. Each RBM layer is made up of a set of visible and hidden units that are trained by maximizing the relevant logarithmic likelihood function. During the pretraining phase, the output of each layer is used as the input to the next layer, allowing the feature representation to be learned layer by layer. The complete network is connected for supervised learning via backpropagation techniques during the fine-tuning phase to further improve network parameters. Figure 3 shows a common DBN network model, which consists of a stack of two RBMs and a back propagation (BP) network stack, with each RBM being a two-layer structure.

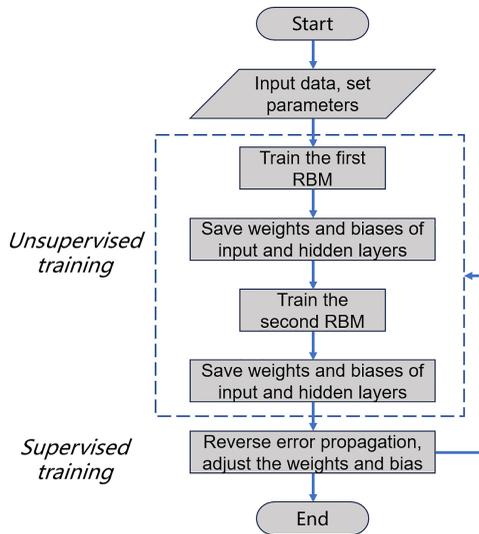
Figure 4 shows the two primary components of the DBN network model training process: unsupervised learning that trains from the bottom to the top layer and supervised learning that trains from the top layer back to the bottom layer.



Source(s): Authors' own work

Prediction of surface roughness

Figure 3. Structure of DBN model



Source(s): Authors' own work

Figure 4. Training process of DBN

In the unsupervised learning process, the RBM uses the greedy algorithm to learn the implicit information of the original data. The DBN network model generally uses the backpropagation algorithm, inputs the final high-level data features obtained from the highest level of the RBM into the BP network, and fine-tunes the entire DBN parameters to achieve the optimal conditions of the network.

DBN has shown exceptional performance in a variety of domains. However, DBN training is a somewhat difficult and computationally intensive procedure that requires layer-by-layer training and fine-tuning of various model parameters. This can lead to lengthier training times and the need for greater computational resources. Furthermore, the model structure, number of layers, number of hidden nodes, and other hyperparameters influence DBN performance. To optimize parameter selection, DBN can be paired with an intelligent optimization method.

2.5 Sparrow search algorithm

The sparrow search algorithm (SSA) is a new swarm intelligence system suggested by Xue and Shen (2020) that is based on the foraging behavior of sparrows. The SSA and other algorithms are comparable, and they are constantly optimized by position updates. The finder location update formula is given by:

$$X_{ij}^{t+1} = \begin{cases} X_{ij}^t \cdot \exp\left(-\frac{i}{\alpha \cdot iter_{max}}\right) & R_2 < ST \\ X_{ij}^t + Q \cdot L & R_2 \geq ST \end{cases} \quad (4)$$

In the formula, t is the current iteration value, an $iter_{max}$ represents the maximum number of iterations. Represents the position information of the i_{th} sparrow in the j_{th} dimension, $j = 1, 2, \dots, c$. α is a random number in the range (0,1]. R_2 is the alert value in the range [0,1]. ST is the security threshold; Q is a random number with normal distribution. L is the matrix with the ones in it. When $R_2 \geq ST$, the finder is in the dangerous area and will fly to the safe area. When $R_2 < ST$, the finder foraged in the safe area (Gai et al., 2021).

The follower always follows the finder, and when the finder starts foraging, the follower position update formula is:

$$X_{ij}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_{worst} - X_{ij}^t}{i^2}\right) & i > \frac{n}{2} \\ X_p^{t+1} + |X_{ij}^t - X_p^{t+1}| \cdot A^+ \cdot L & i \leq \frac{n}{2} \end{cases} \quad (5)$$

In the formula, X_p is the best position of the producer, X_{worst} represents the global worst position in the current iteration, and A represents the matrix of dimension C whose elements are 1 or -1 and satisfies $A^+ = A^T(AA^T)^{-1}$. When $i > n/2$, it indicates that the follower is hungry and will fly to other locations, and the other followers will forage near the finder's optimal location.

Sparrows that detect early warning are mainly used to protect the population from foraging. The initial position of the reconnaissance warning is randomly generated in the population and the specific position update formula is as follows:

$$X_{ij}^{t+1} = \begin{cases} X_{best}^t + \lambda |X_{ij}^t - X_{best}^t| & f_i \neq f_g \\ X_{ij}^t + K \left(\frac{|X_{ij}^t - X_{worst}^t|}{(f_i - f_w) + \epsilon} \right) & f_i = f_g \end{cases} \quad (6)$$

In the formula, X_{best}^t is the global optimal position; Both β and K are random numbers, but β is normally distributed. f_i is the fitness value of the current sparrow; f_g is the global best fitness; f_w is the global worst fitness value; ϵ is a very small constant; When $f_i >$, it means that the current sparrow is threatened and will update to the current optimal position; when $f_i = f_g$, it means that the sparrow in the optimal position is threatened and needs to update its position.

2.6 Tent chaotic mapping

Many algorithm researchers will employ randomly generated variables during population initialization to ensure the method's optimization efficiency. Because of the features of chaotic

mapping variables, they can distribute variables equally in the search space during algorithm setup, which greatly improves the algorithm's optimization efficiency and accuracy. Logistic mapping is the most popular chaotic mapping in the literature, yet the chance of variables in this mapping being on both sides of [0,1] is higher, resulting in an uneven distribution of variables and low optimization effectiveness.

It is demonstrated that Tent chaotic mapping has higher ergodic and convergence efficiency than logistic mapping. This paper employs Tent chaotic mapping (Li *et al.*, 2020), with the following formula:

$$z_{k+1} = \begin{cases} \frac{z_k}{\beta}, z_k \in (0, \beta) \\ \frac{(1 - z_k)}{(1 - \beta)}, z_k \in (\beta, 1) \end{cases} \quad (7)$$

The value of in the formula is between [0,1], and the appropriate value can be chosen based on the unique situation, and the value of in this paper is 0.7. In general, the first value of the sequence is still created by a random function and iterated by a mapping formula.

Tent chaotic sequence has small periods and is unstable. In order to prevent it from falling into periodic points without affecting the three major characteristics of chaotic variables, based on the random variable $rand(0, 1) \cdot \frac{1}{N_T}$. Tent mapping expression is as follows:

$$z_{k+1} = \begin{cases} \frac{z_k}{\beta} + rand(0, 1) \cdot \frac{1}{N_T} & 0 \leq z \leq \beta \\ \frac{(1 - z_k)}{(1 - \beta)} + rand(0, 1) \cdot \frac{1}{N_T} & \beta < z \leq 1 \end{cases} \quad (8)$$

In the formula, N_T is the number of particles in the chaotic sequence; and (0,1) is a random number.

The steps to produce a chaotic sequence are as follows:

- (1) Randomly generate the initial value z in (0,1), denoting $i = 0$;
- (2) Perform iteration to generate Z sequence, I increment by 1;
- (3) If the maximum number of iterations is reached, the program stops running and the Z sequence is saved.

Using Tent chaos mapped data as the initial population location information, the algorithm's search diversity is preserved and the individual's ability to jump out of the local optimal in the search process is improved, as is the algorithm's convergence speed and global search optimal ability. The process of applying Tent-SSA optimization algorithm to search DBN hyperparameters is shown in Figure 5.

3. Experimental setup

This work created a multi-sensor fusion experiment platform powered by physical signals and CNC machining parameters data, as shown in Figure 6. The Dashan EUP series D10 milling cutter and 6,061 aluminum workpiece are applied on the high-speed direct-drive machining center FGV1060L. The workpiece is installed on the dynamometer, which is fixed on the machine table of FGV1060L. The FGV1060L has a maximum feed speed of 30m/min in all X/Y/Z directions and a maximum spindle speed of 24,000 RPM (revolutions per minute), which makes it suitable for high-speed machining.

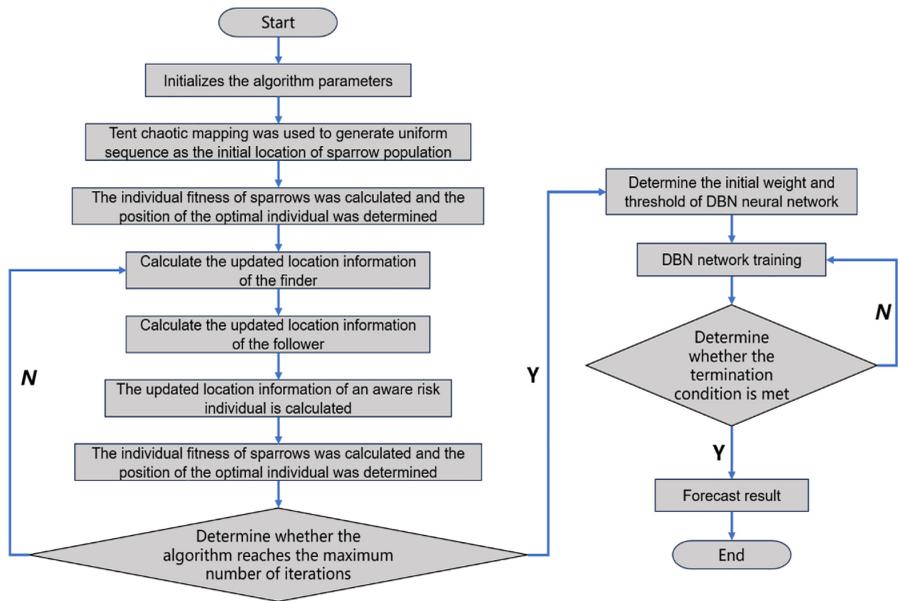


Figure 5. The optimization process of chaotic Tent-SSA-DBN neural network

Source(s): Authors' own work

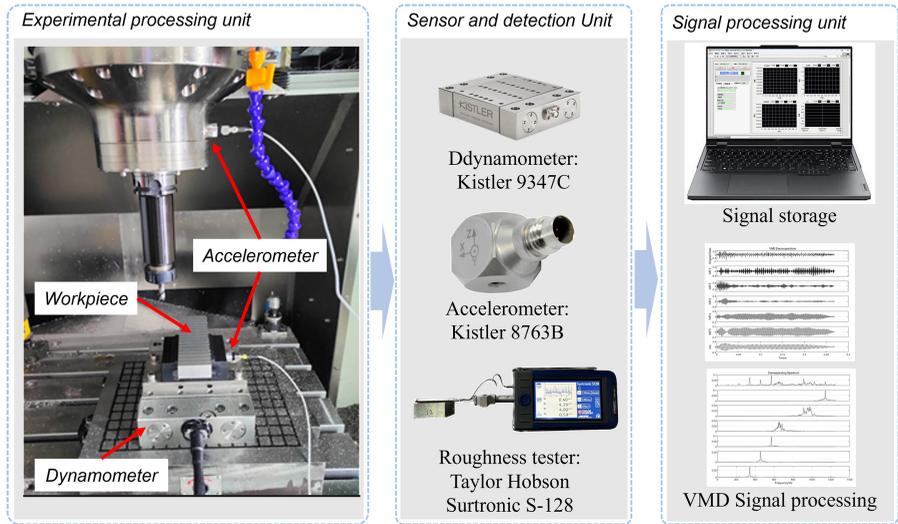


Figure 6. Diagram of experimental setup

Source(s): Authors' own work

The signal data collected by the signal acquisition unit were analyzed and processed using statistical and informatics methods, and the signal features that could reflect the dynamic changes of the cutting process, and the cutting process monitoring was completed using the neural network model. The testing included single-factor side milling, orthogonal side

milling, single-factor face milling and orthogonal face milling. To establish an appropriate parameter range, the experimental speed range of 3,000–15,000 (r/min), feed speed range of 1,000–22,000 (mm/min), cutting width and depth of cut by the features of face milling or side milling.

4. Results and discussions

4.1 Data preprocessing

(1) Input data shape

The number of features that can be extracted from samples for force signals and vibration signals respectively is shown in Table 3. The workpiece surface roughness prediction model's input comprises a total of $26 \times 6 + 8 \times 2 \times 3 + 6 \times 2 \times 3 = 240$ feature dimensions.

(2) One-hot encoding

One-hot coding is a feature coding approach that is commonly used to convert discrete variables to numerical variables. The original discrete variables are turned into a set of numerical variables that are easier to deal with in data analysis and ML by using one-hot coding. The concept of distance measures (such as Euclidean distance) is included in the method by one-hot encoding, allowing the model to better reflect the similarity between variables.

There are four roughness eigenvalues, and one-hot encoding is employed to transform them to 0/1 numeric variables. For example, in the case of 2, 1, 4, the values after encoding are [0, 1, 0, 0], [0, 1, 0, 0], and [0, 0, 0, 1].

(3) Data oversampling

There are certain machining criteria for component surface roughness in the actual production process. As a result, this study combines the Ra values with the complementary values of the Ra series and splits the Ra values (1–3.2 μm) into four parts referred to as stages 1–4. As illustrated in Figure 7, each stage is classified and the roughness grades are determined using the DBN model. Using the model described in this study, it is simple to assess whether a product's surface roughness falls within the desired roughness class, allowing for more efficient production of components that meet machining precision while minimizing the time cost of repeated steps to measure surface roughness.

All surface roughness data samples were divided into four groups, each with 17, 30, 22 and 12 data samples.

The data's 2D distribution is investigated for roughness-related characteristics, and it is discovered that the distribution of the features is characterized by an uneven distribution and a gap in the number of features. Table 4 shows the quantity of data in each class before and after oversampling using the ADASYN and SMOTE algorithms on the original data. The scatter plots of the different types of data obtained by using these two oversampling methods are shown in Figure 8.

Signal	Time/Frequency/time-frequency domain characteristics (per direction)	VMD decomposition features (per direction)
Force signal (three directions)	26	16
Vibration signal (three directions)	26	12

Source(s): Authors' own work

Table 3. The number of features that can be extracted from a sample

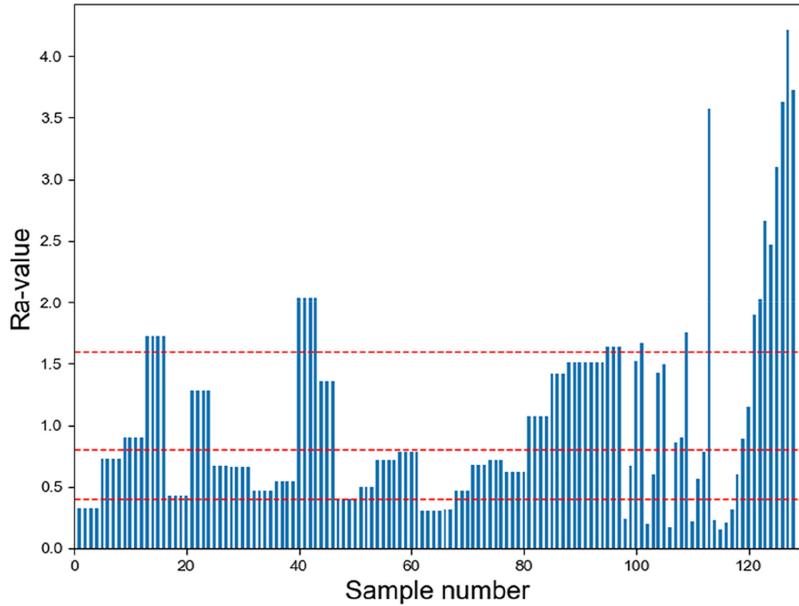


Figure 7.
Roughness and classification obtained by experiment

Source(s): Authors' own work

Table 4.
The number of data samples for each category of original data and oversampled data

Level	Ra/ μm	Original	Smote	ADASYN
1	<0.2	18	36	37
2	0.4~0.8	56	112	111
3	0.8~1.6	32	64	69
4	>1.6	22	44	43

Source(s): Authors' own work

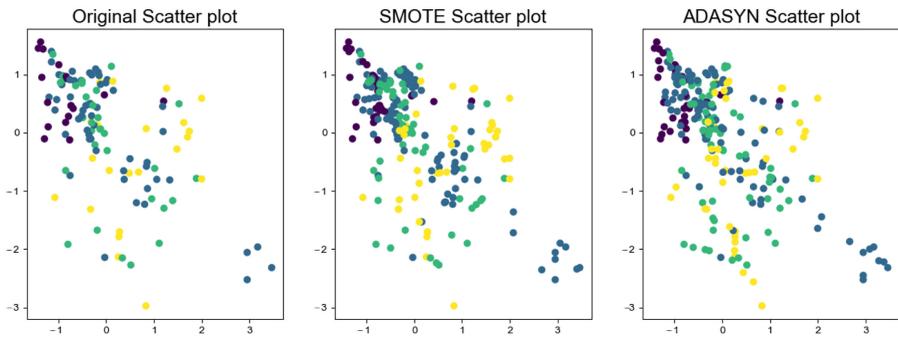


Figure 8.
Comparison of data sets with different oversampling methods

Source(s): Authors' own work

ADASYN expands the data more consistently and more for the regions with less data distribution, On the contrary, the SMOTE algorithm's data expansion is more concentrated in the region where the data is spread, making it susceptible to overfitting for data classification. This is due to the trained model's difficulty distinguishing between the few samples in the

initial dataset, and it can help to address the imbalance issue in the network data and improve the model's generalizability. As a result, ADASYN oversampled data is used to build the roughness prediction model.

(4) Normalization

Data standardization, a common data preprocessing method that transforms data into a normal distribution with a mean of zero and a standard deviation of one, can be used to solve problems caused by large gaps between feature data of different dimensions in a dataset, such as slow model training and insignificant improvement in accuracy.

The data preprocessing method was used to solve problems caused by large gaps between feature data of different dimensions in a dataset. That normalization process transforms data into a normal distribution with a mean of zero and a standard deviation of one, the transformation formula for StandardScaler is as follows:

$$x_0 = \frac{x - \mu}{\sigma} \tag{9}$$

Where x represents the original data, x_0 represents the converted data, μ denotes the mean value of the original data, σ denotes the standard deviation of the original data.

4.2 Classification prediction

(1) Prediction of baseline DBN categorization

Set the DBN model to have two hidden layers of 200 and 50 nodes, triggered with the ReLU function and a random number seed of 7. Figure 9a depicts the training recognition results, whereas Figure 9b depicts the confusion matrix. It is discovered that 49 of the 52 points in the test set are right, resulting in an accuracy of 94.23%.

(2) Results comparison for other input data

To compare the findings, this work continues to fit the input data and roughness data without changing the baseline DBN model parameters and simply adjusting the input feature dimensions and sample size. The three input data dimensions in this section are:

- The input is 128 samples and 244 features with four-dimensional cutting parameters (spindle speed, feed rate, width of cut, depth of cut) and without ADASYN oversampling.

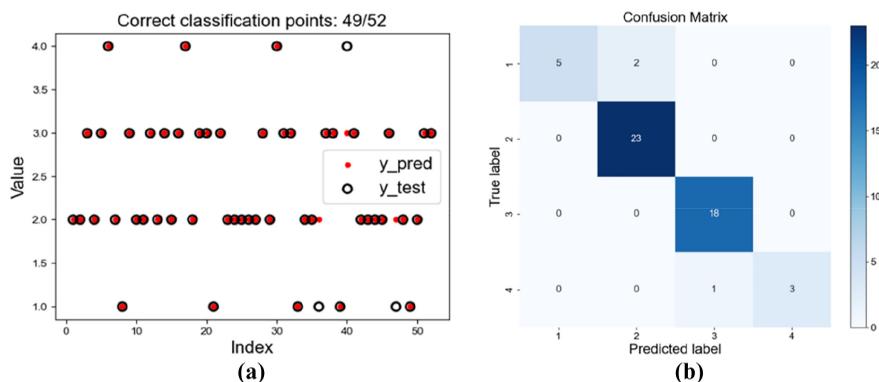


Figure 9. Baseline DBN classification prediction result

Source(s): Authors' own work

- The input is 128 samples and 240 features without four-dimensional cutting settings and ADASYN oversampling.
- The input is 260 samples and 244 features when using four-dimensional cutting settings with ADASYN oversampling.

Figure 10 depicts the test set’s classification prediction results. ADASYN oversampling has a greater performance improvement for the roughness classification prediction model than the results of the input parameter model (the accuracy increases from 80.77% to 94.23%). Still, the cutting parameter, which is highly correlated with roughness, does not significantly improve the prediction model’s accuracy.

(3) Improvement of the DBN classification model with Tent-SSA

Tent-SSA is used to explore four parameters of the DBN classification model: node number of two hidden layers, learning rate and random number seed while. The first hidden layer number interval is [50,400], the second hidden layer number interval is [10,100], the learning rate interval is $[1 \times 10^{-7}, 1]$, and the random number seed interval is $[-10,000, 10,000]$. The Tent-SSA intelligent optimization algorithm’s population number is set to 80. The maximum number of iterations is set to 10, and the fitness function’s return value is the DBN model’s error rate on the test set. Figure 11 indicates the optimization procedure. The final optimization result is as follows: the first hidden layer has 261 nodes, the second hidden layer has 61 nodes, the learning rate is 0.009174 and the random number seed is 0. Figure 12

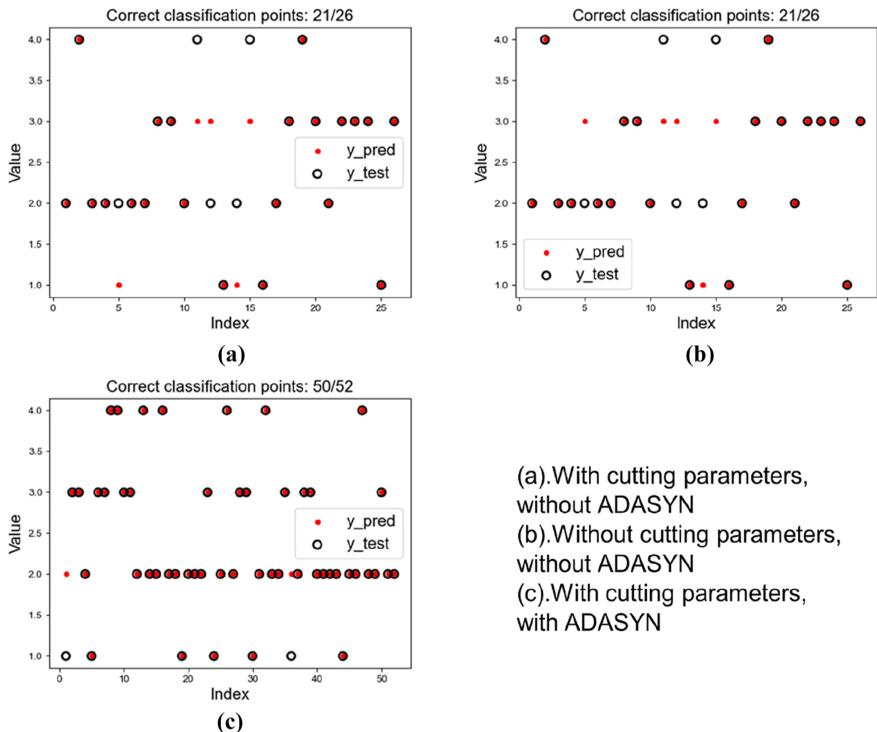
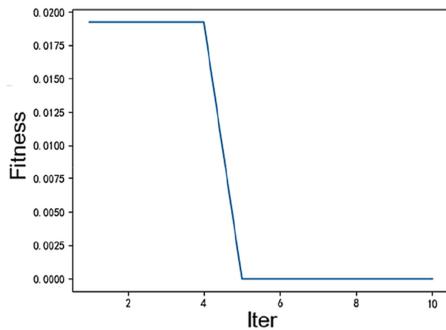
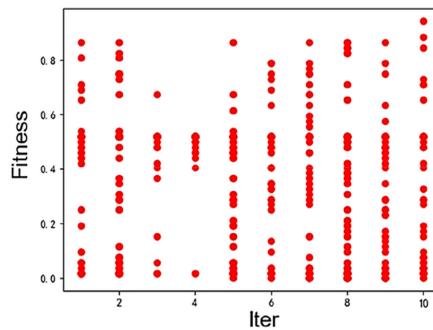


Figure 10. Prediction results for three different input dimensions

Source(s): Authors’ own work

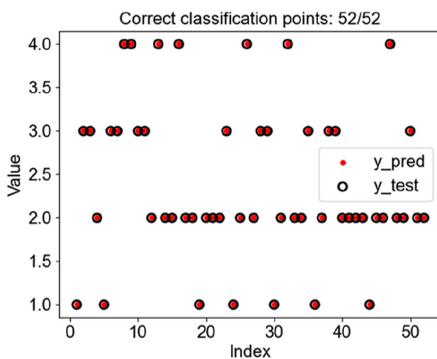


Source(s): Authors' own work



Prediction of surface roughness

Figure 11. Tent-SSA optimization process



Source(s): Authors' own work

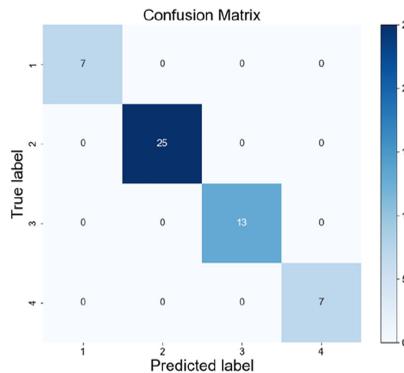


Figure 12. Optimization of DBN classification prediction results

indicates the model classification prediction results produced by training these DBN parameters. All 52 classification predictions in the test set are correct.

4.3 Regression prediction

(1) Prediction of baseline DBN regression

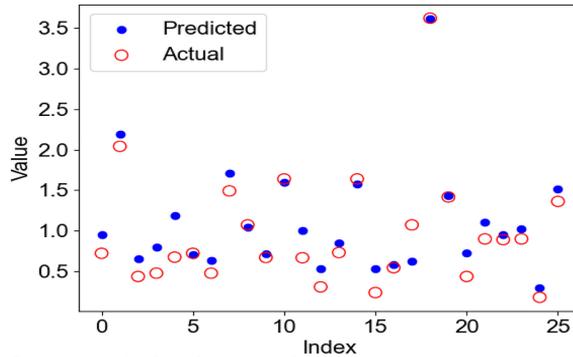
Set the DBN model to include two hidden layers, 200 and 50 nodes, activated by the ReLU function and a random number seed of 7. It is shown in Figure 13 that the mean absolute error (MAE) was 0.1701, the coefficient of determination (R2) was 0.9071 and the maximum absolute error (MaxAE) was 0.5157.

(2) Improvement of the DBN regression model with Tent-SSA

- Using root-mean-square error (RMSE) as a fitness function

Tent-SSA was used to explore four parameters of the DBN regression model: node number of two hidden layers, learning rate and random number seed. The first hidden layer number interval is [50,400], and the second hidden layer number interval is [10,100], the learning rate interval is $[1 \times 10^{-7}, 1]$ and the random number seed interval is $[-10,000, 10,000]$. The tent-SSA intelligent optimization algorithm's population number is set to 80. The maximum

Figure13.
Baseline DBN
regression prediction
result



Source(s): Authors' own work

number of iterations is set to ten, and the fitness function's return value is the RMSE of the DBN model on the test set. Figure 14 indicates the optimization procedure. The final optimization result is that the first hidden layer has 78 nodes, the second hidden layer has 75 nodes, the learning rate is 0.025446 and the random number seed is 443. Figure 15 is the model classification prediction results produced by training these DBN parameters. The calculated RMSE for the regression prediction of 26 points in the test set is 0.1264, the MAE is 0.0934 and the coefficient of determination (R^2) is 0.9682. The absolute MaxAE is 0.4149, the optimization effect is favorable.

- Using MaxAE as a fitness function

When compared to root mean square (RMS) error, adopting maximum absolute error as the fitness function MaxAE has several advantages, including:

- Robustness: Because MaxAE is unaffected by outliers, it is more robust than RMSE for data sets that may contain outliers or noise.
- Simplicity: In contrast to the complex calculations required by RMSE, MaxAE is a simple and understandable metric that directly estimates the biggest divergence between predicted and true values.
- Interpretability: MaxAE has a clear physical meaning that is simple to explain and grasp, signifying the greatest deviation that can occur during the prediction process of the model.

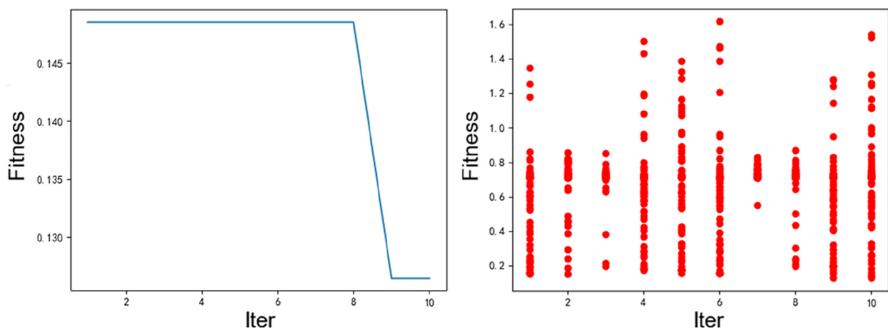


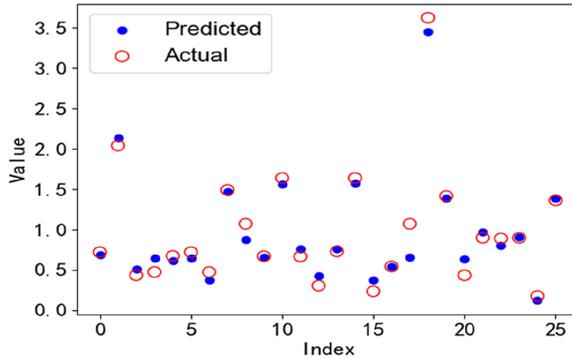
Figure 14.
Tent-SSA optimization
process (base
on RMSE)

Source(s): Authors' own work

- Faster convergence: Because MAE is more sensitive to variations between predicted and true values, it may result in faster convergence during training.

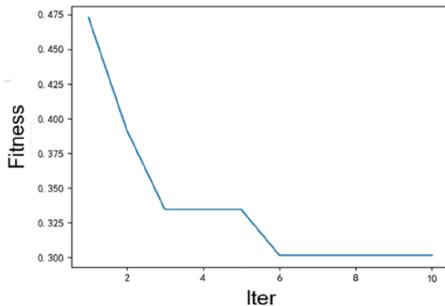
Prediction of surface roughness

Figure 16 shows the optimization procedure, the first hidden layer has 328 nodes, the second hidden layer has 47 nodes, the learning rate is 0.009279, and the random number seed is -1,116. As the model classification prediction results produced by training these DBN parameters in Figure 17, for the regression prediction of 26 points in the test set, the obtained



Source(s): Authors' own work

Figure 15. Optimization of DBN regression prediction results (base on RMSE)



Source(s): Authors' own work

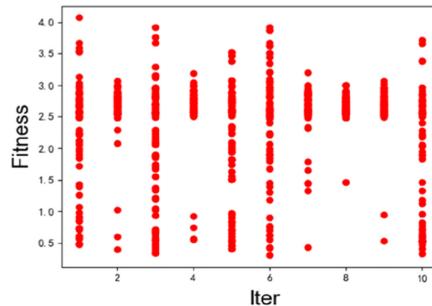
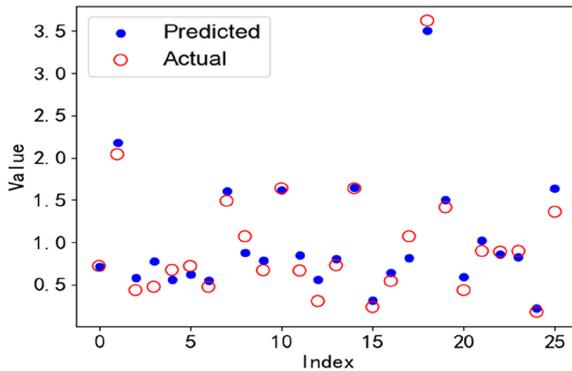


Figure 16. Tent-SSA optimization process (base on MaxAE)



Source(s): Authors' own work

Figure 17. Optimization of DBN regression prediction results (base on MaxAE)

RMSE is 0.1459, the MAE is 0.1228 and the coefficient of determination (R^2) is 0.9576. The maximum absolute error MaxAE is 0.3013, the optimization obtains a good effect.

4.4 Optimized model parameters and results

The training settings and outcomes acquired in this paper are provided in Table 5, as can be observed from the above training process description:

To summarize, the method suggested in this research is viable and can effectively forecast machining results by monitoring machining process information, that is, surface roughness classification and regression. At the same time, the suggested method's validity and superiority are demonstrated by comparing it to the prediction results of the DBN model without Tent-SSA optimization.

5. Summary and prospects

5.1 Conclusion

The signals generated by the machine tool during the machining process contain a lot of useful information, and the relevant features can be extracted by real-time monitoring and processing of the signals, after which the DBN model can be trained to effectively predict surface roughness classification and regression. A sparrow search technique based on Tent chaotic mapping is utilized to achieve the optimal combination of DBN model parameters, and different fitness functions are used in different applications. The modified model is then applied to roughness prediction, which fully validates the suggested method's effectiveness. Finally, the following conclusions are reached:

- (1) For real-time monitoring signals during machine tool machining, feature extraction, time-frequency domain analysis and VMD decomposition can greatly reflect vital

Classification prediction model			
Model		Reference DBN	Tent-SSA-DBN
Number of nodes in the 1st hidden layer		200	261
Number of nodes in the 2nd hidden layer		50	61
Learning rate		0.01	0.009174
Random seed		7	0
Accuracy		94.23%	99.99%
Regression forecasting model			
Model	Reference DBN	Tent-SSA-DBN (Based on RMSE)	Tent-SSA-DBN (Based on MaxAE)
Number of nodes in the 1st hidden layer	200	78	328
Number of nodes in the 2nd hidden layer	50	75	47
Learning rate	0.01	0.025446	0.009279
Random seed	7	443	-1,116
RMSE	0.2161	0.1264	0.1459
MAE	0.1701	0.0934	0.1228
R^2	0.9070	0.9682	0.9576
MaxAE	0.5157	0.4149	0.3013
Optimization effect	-	41.51%	41.57%

Table 5. Optimum model parameters and results

Source(s): Authors' own work

information, creating suitable conditions for neural network model training. According to comparisons, cutting parameters, despite being highly connected with roughness, do not greatly increase the accuracy of the prediction model in the DBN model.

- (2) The roughness dataset from the experiments was split into four classes, and the original dataset was expanded using the ADASYN algorithm dataset to approximately twice its original size for classification prediction. This resulted in a significant increase in the classification accuracy for the baseline DBN model, which went from 80.77% to 94.23%.
- (3) A DBN model based on the Tent-improved SSA is proposed to classify and predict surface roughness. Tent-SSA searches the optimal weight parameters of the DBN model, reducing the uncertainty factors of the interference prediction model and increasing the classification accuracy by 5.77% based on ADASYN optimization.
- (4) When the baseline DBN model is used to directly forecast roughness, the RMSE and maximum absolute error (MaxAE) are 0.2161 and 0.5157, respectively. When the goal function is RMSE, using Tent-SSA to search for the optimal weight parameters of the DBN model, the RMSE can be reduced to 0.1264 and the MaxAE can be reduced to 0.3013. When RMSE is the objective function, the RMSE can be decreased to 0.1264; when MaxAE is the objective function, the MaxAE may be reduced to 0.3013. The optimization effect reaches more than 40% based on the various practical demands to optimize the model parameters.

In conclusion, this paper proposes an enhanced deep learning and data augmentation approach for surface roughness prediction. This approach successfully increases the accuracy of surface roughness prediction. It can be used for real-time quality prediction during the machining process. Applying the method, rapid process diagnosis, abnormal early warning and dynamic control of cutting parameters can be realized, which makes a certain contribution to the transformation of machine tool processing to intelligent systematization.

5.2 Prospects

- (1) The roughness regression prediction model is optimized using two fitness functions. These functions are based on the root mean square error and the greatest absolute error. Each strategy has advantages and disadvantages and should be used based on the circumstances. It could be conceivable to develop a new fitness function that takes into account both RMS error and maximum absolute error. This new function might be tailored to the user's specific requirements by altering the coefficient.
- (2) The addition of four cutting parameters did not significantly improve the model's prediction effect, indicating that the influence of cutting parameters on roughness is reflected in the real-time monitoring signal. However, whether the roughness differences induced by different material qualities and tool parameters may also be properly reflected by the machine tool processing's real-time monitoring signal requires additional experimental research.
- (3) It is certain that the model in this paper can be extended to the research of grinding, turning, and other processing technologies, by establishing the evaluation criteria based on a sufficient number of experimental samples.
- (4) While the workpiece and tool materials used in this paper's machining state monitoring research are the same, future research should select a wide range of

materials for numerous cutting experiments on the workpiece and tool to further demonstrate the applicability of the monitoring technique described in this paper.

- (5) The neural network model is built on feature data which is easily interpreted and has a straightforward structure, while the features only contained rudimentary information. The subsequent research can take the original signal data or the two-dimensional image data transformed by the original signal as the input of the neural network model. Adaptive feature extraction and prediction will be carried out in the deep neural network model structure using 1D CNN, 2D CNN, recurrent neural network (RNN) and other model structures.

References

- Abu-Mahfouz, I., El Ariss, O., Esfakur Rahman, A.H.M. and Banerjee, A. (2017), "Surface roughness prediction as a classification problem using support vector machine", *The International Journal of Advanced Manufacturing Technology*, Vol. 92 No. 1, pp. 803-815, doi: [10.1007/s00170-017-0165-9](https://doi.org/10.1007/s00170-017-0165-9).
- Agrawal, A., Goel, S., Rashid, W.B. and Price, M. (2015), "Prediction of surface roughness during hard turning of AISI 4340 steel (69 HRC)", *Applied Soft Computing*, Vol. 30, pp. 279-286, doi: [10.1016/j.asoc.2015.01.059](https://doi.org/10.1016/j.asoc.2015.01.059).
- Ahmed, G., Er, M.J., Fareed, M.M.S., Zikria, S., Mahmood, S., He, J., Asad, M., Jilani, S.F. and Aslam, M. (2022), "DAD-net: classification of alzheimer's disease using ADASYN oversampling technique and optimized neural network", *Molecules, Multidisciplinary Digital Publishing Institute*, Vol. 27 No. 20, p. 7085, doi: [10.3390/molecules27207085](https://doi.org/10.3390/molecules27207085).
- Benardos, P.G. and Vosniakos, G.C. (2002), "Prediction of surface roughness in CNC face milling using neural networks and Taguchi's design of experiments", *Robotics and Computer-Integrated Manufacturing*, Vol. 18 No. 5, pp. 343-354, doi: [10.1016/S0736-5845\(02\)00005-4](https://doi.org/10.1016/S0736-5845(02)00005-4).
- Chen, Y., Sun, R., Gao, Y. and Leopold, J. (2017), "A nested-ANN prediction model for surface roughness considering the effects of cutting forces and tool vibrations", *Measurement*, Vol. 98, pp. 25-34, doi: [10.1016/j.measurement.2016.11.027](https://doi.org/10.1016/j.measurement.2016.11.027).
- Cooper, C., Zhang, J., Guo, Y.B. and Gao, R.X. (2023), "Surface roughness prediction through GAN-synthesized power signal as a process signature", *Journal of Manufacturing Systems*, Vol. 68, pp. 660-669, doi: [10.1016/j.jmsy.2023.05.016](https://doi.org/10.1016/j.jmsy.2023.05.016).
- Dragomiretskiy, K. and Zosso, D. (2014), "Variational mode decomposition", *IEEE Transactions on Signal Processing, Presented at the IEEE Transactions on Signal Processing*, Vol. 62 No. 3, pp. 531-544, doi: [10.1109/TSP.2013.2288675](https://doi.org/10.1109/TSP.2013.2288675).
- Fernandez, A., Garcia, S., Herrera, F. and Chawla, N.V. (2018), "SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary", *Journal of Artificial Intelligence Research*, Vol. 61, pp. 863-905, doi: [10.1613/jair.1.11192](https://doi.org/10.1613/jair.1.11192).
- Gai, J., Zhong, K., Du, X., Yan, K. and Shen, J. (2021), "Detection of gear fault severity based on parameter-optimized deep belief network using sparrow search algorithm", *Measurement*, Vol. 185, 110079, doi: [10.1016/j.measurement.2021.110079](https://doi.org/10.1016/j.measurement.2021.110079).
- Guo, W., Wu, C., Ding, Z. and Zhou, Q. (2021), "Prediction of surface roughness based on a hybrid feature selection method and long short-term memory network in grinding", *The International Journal of Advanced Manufacturing Technology*, Vol. 112 No. 9, pp. 2853-2871, doi: [10.1007/s00170-020-06523-z](https://doi.org/10.1007/s00170-020-06523-z).
- Guo, M., Zhou, J., Li, X., Lin, Z. and Guo, W. (2023), "Prediction of surface roughness based on fused features and ISSA-DBN in milling of die steel P20", *Scientific Reports*, Vol. 13 No. 1, 15951, doi: [10.1038/s41598-023-42968-4](https://doi.org/10.1038/s41598-023-42968-4).
- He, K., Xu, Q. and Jia, M. (2015), "Modeling and predicting surface roughness in hard turning using a Bayesian inference-based HMM-SVM model", *IEEE Transactions on Automation Science and*

-
- Engineering, Presented at the IEEE Transactions on Automation Science and Engineering*, Vol. 12 No. 3, pp. 1092-1103, doi: [10.1109/TASE.2014.2369478](https://doi.org/10.1109/TASE.2014.2369478).
- Hinton, G.E. and Salakhutdinov, R.R. (2006), "Reducing the dimensionality of data with neural networks", *Science (New York, N.Y.)*, Vol. 313 No. 5786, pp. 504-507, doi: [10.1126/science.1127647](https://doi.org/10.1126/science.1127647).
- Kong, D., Zhu, J., Duan, C., Lu, L. and Chen, D. (2020), "Bayesian linear regression for surface roughness prediction", *Mechanical Systems and Signal Processing*, Vol. 142, 106770, doi: [10.1016/j.ymssp.2020.106770](https://doi.org/10.1016/j.ymssp.2020.106770).
- Li, Y., Han, M. and Guo, Q. (2020), "Modified whale optimization algorithm based on tent chaotic mapping and its application in structural optimization", *KSCIE Journal of Civil Engineering*, Vol. 24 No. 12, pp. 3703-3713, doi: [10.1007/s12205-020-0504-5](https://doi.org/10.1007/s12205-020-0504-5).
- Lin, W.-J., Lo, S.-H., Young, H.-T. and Hung, C.-L. (2019), "Evaluation of deep learning neural networks for surface roughness prediction using vibration signal analysis", *Applied Sciences*, Vol. 9 No. 7, p. 1462, doi: [10.3390/app9071462](https://doi.org/10.3390/app9071462).
- Lu, J., Liao, X., Li, S., Odsuyang, H., Chen, K. and Huang, B. (2019), "An effective ABC-SVM approach for surface roughness prediction in manufacturing processes", *Complexity*, Vol. 2019, pp. 1-13, doi: [10.1155/2019/3094670](https://doi.org/10.1155/2019/3094670).
- Lv, J., Huang, X., Zhu, J. and Zhang, Z. (2021), "An end-to-end deep learning model to predict surface roughness", in Han, Q., McLoone, S., Peng, C. and Zhang, B. (Eds), *Intelligent Equipment, Robots, and Vehicles*, Springer, Singapore, pp. 607-616, doi: [10.1007/978-981-16-7213-2_58](https://doi.org/10.1007/978-981-16-7213-2_58).
- Pan, Y., Kang, R., Dong, Z., Du, W., Yin, S. and Bao, Y. (2022), "On-line prediction of ultrasonic elliptical vibration cutting surface roughness of tungsten heavy alloy based on deep learning", *Journal of Intelligent Manufacturing*, Vol. 33 No. 3, pp. 675-685, doi: [10.1007/s10845-020-01669-9](https://doi.org/10.1007/s10845-020-01669-9).
- Scarpiniti, M., Colasante, F., Tanna, S., Ciancia, M., Lee, Y. and Uncini, A. (2021), "Deep Belief Network based audio classification for construction sites monitoring", *Expert Systems with Applications*, Vol. 177, 114839, doi: [10.1016/j.eswa.2021.114839](https://doi.org/10.1016/j.eswa.2021.114839).
- Xue, J. and Shen, B. (2020), "A novel swarm intelligence optimization approach: sparrow search algorithm", *Systems Science and Control Engineering*, Vol. 8 No. 1, pp. 22-34, doi: [10.1080/21642583.2019.1708830](https://doi.org/10.1080/21642583.2019.1708830).
- Zhou, T., He, L., Wu, J., Du, F. and Zou, Z. (2019), "Prediction of surface roughness of 304 stainless steel and multi-objective optimization of cutting parameters based on GA-GBRT", *Applied Sciences*, Vol. 9 No. 18, p. 3684, doi: [10.3390/app9183684](https://doi.org/10.3390/app9183684).

Corresponding author

Miaoxian Guo can be contacted at: guomx@usst.edu.cn

For instructions on how to order reprints of this article, please visit our website:

www.emeraldgrouppublishing.com/licensing/reprints.htm

Or contact us for further details: permissions@emeraldinsight.com