http://dx.doi.org/10.62762/TSCC.2025.978286

RESEARCH ARTICLE

RUL Prediction of the Injection Lance in Copper Top-Blown Smelting Using KPCA and TSO-Optimized LSSVM

Qiang Peng[1], **Gengen Li**[2,3], **Chunxi Yang**[2,3,*], **Xiufeng Zhang**[2,3], **Jing Na**[2,3] **and Mou Li**[2,3]

[1] Huize Smelting Branch, Yunnan Chihong Zinc and Germanium Co., Ltd., Huize 654200, China
[2] Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming 650500, China
[3] Yunnan Key Laboratory of Intelligent Control and Application, Kunming University of Science and Technology, Kunming 650500, China

Abstract

As the core component of the copper top-blown smelting, the service life of the injection lance critically affects production stability. To monitor the operating condition of the injection lance, a data-driven model is proposed to predict the Remaining Useful Life (RUL) or service life, namely, the DKT-LSSVM model. Firstly, to reduce noise interference, the Daubechies wavelet with four vanishing moments (DB4) denoising is used to process the raw data. Then, the Kernel Principal Component Analysis (KPCA) method is utilized to extract the principal components from the denoised data, which retains at least 90% information content (18 principal components are obtained). These principal components are used as inputs to a Least Square Support Vector Machine (LSSVM) model to predict the RUL of the injection lance, and a Tuna Swarm Optimization (TSO) algorithm is proposed to optimize the hyperparameters of the LSSVM. The results show that the proposed algorithm performs well in RUL prediction of the injection lance, with RMSE=1.2274 (day), MAE=0.6623 (day) and R^2=0.9308. Therefore, the proposed algorithm can provide effective RUL prediction for the injection lance, reduce its operational risks, and improve the stability and reliability of the copper top-blown smelting system.

Keywords: injection lance, least square support vector machine (LSSVM), tuna swarm optimization (TSO) algorithm, remaining useful life (RUL).

1 Introduction

Conventional copper concentrate smelting techniques are being replaced by new copper smelting processes, due to the high energy consumption and serious environmental pollution [1]. In many advanced copper pyrometallurgy processes, the Isa smelting process is widely used, because of its low required investment, rapid smelting, wide raw material adaptability, high productivity, and low energy consumption [2]. In the Isa smelting process,

Submitted: 28 October 2025
Accepted: 26 November 2025
Published: 30 November 2025

Vol. 2, **No.** 4, 2025.
10.62762/TSCC.2025.978286

***Corresponding author:**
✉ Chunxi Yang
ycx@kmust.edu.cn

Citation
Peng, Q., Li, G., Yang, C., Zhang, X., Na, J., & Li, M. (2025). RUL Prediction of the Injection Lance in Copper Top-Blown Smelting Using KPCA and TSO-Optimized LSSVM. *ICCK Transactions on Sensing, Communication, and Control*, 2(4), 238–249.

oxygen-enriched air is vertically injected into the high-temperature molten bath by the injection lance and then forms bubbles to stir the latter [3]. The injection lance is thus regarded as a key component of the Isa smelting system. Therefore, the injection lance is of importance for enhancing the momentum and heat transfer inside the molten phase.

However, the service life of the injection lance is severely affected by the work process (high stirring intensity). Typically, the injection lance fails owing to bending, tip burning, body wear, plugging, tip bogging, or accretion, among which bending and tip burning are considered to be the main reasons for lance damage [4]. Once the injection lance fails, it will significantly reduce the operating efficiency of the smelting process. Thus, accurately predicting the Remaining Useful Life (RUL) of the injection lance is crucial for its health management, such as scheduling maintenance activities, optimizing resources, and minimizing unplanned failures.

It is well known that datasets are very important for modeling. Due to constraints, the collected operational data of the injection lance are very limited. Therefore, data processing and model selection are particularly important. The dataset used in this paper has three main characteristics: 1) Noise interference; 2) Feature redundancy; 3) Few-shot data. These characteristics further increase the difficulty of modeling.

In view of the above considerations, a DKT-LSSVM algorithm is developed in this paper to predict the RUL of the injection lance. In this algorithm, the Daubechies wavelet with four vanishing moments (DB4) denoising is utilized to reduce noise in the raw data. Then, the Kernel Principal Component Analysis (KPCA) is introduced to eliminate the correlation between the features and to realize the dimensionality reduction process. In light of the superiority of the Least Square Support Vector Machine (LSSVM) model, it is used to model the relationship between the principal components and the RUL of the injection lance. In addition, the Tuna Swarm Optimization (TSO) algorithm is used to optimize the hyperparameters of the LSSVM model. Finally, the proposed method is validated using actual operation data on the injection lance from a copper smelter. The results show that the proposed algorithm has the best evaluation metrics: RMSE=1.2274, MAE=0.6623, R^2=0.9308. The main contributions of this paper are summarized as follows:

1. We constructed a DKT-LSSVM model, which combines the LSSVM model, DB4 denoising technology, KPCA method and TSO algorithm. The DB4 denoising and KPCA are used to enhance and extract features from raw data as modeling input. Meanwhile, the TSO is introduced to optimize the hyperparameters of the LSSVM model.
2. The effectiveness and superiority of the proposed DKT-LSSVM RUL assessment model are validated by a case study. The research results offer a novel and effective methodology in the field of the RUL assessment of injection lance, which is expected to provide research idea for the health management of the injection lance.

The remainder of this paper is organized as follows: Related work is introduced in Section 2. Section 3 describes the data preprocessing, which includes the DB4 denoising and the data normalization. In Section 4, the methodology is presented, including the KPCA, LSSVM model and TSO algorithm. Section 5 is a case study. Finally, conclusions are presented in Section 6.

2 Related Work

With the development of artificial intelligence [5], the development of machine learning [6] and other technologies have brought new opportunities for research on the RUL prediction for the injection lance. The data-driven RUL prediction methods have garnered significant attention [7], primarily because they are independent of detailed physical modeling and extensive prior domain knowledge [8]. The data-driven methods directly learn the mapping from monitoring data to RUL, enabling end-to-end RUL prediction. Especially, Deep Learning (DL) models have gained prominence in the field of RUL prediction, which enhances prediction performance through their powerful learning and representation of degraded information [9, 10]. Song *et al*. [11] presented a deep learning-based RUL prediction method, which uses an attention mechanism to weight sequence data representations. Zhang *et al*. [12] combined Convolutional Neural Network (CNN) with Recurrent Neural Network (RNN) to estimate RUL. The Long Short-Term Memory (LSTM) was also used by Zheng *et al*. [13] and Zhang *et al*. [14] to predict RUL. To make full use of the sensor data sequence bidirectionally, Wang *et al*. [15] proposed a new data-driven approach with Bidirectional Long Short-Term Memory (BiLSTM) network for RUL estimation.

Unfortunately, DL models typically require a large amount of training data to ensure their performance.

Moreover, data collection proves both costly and time-consuming in many industrial scenarios [16]. Therefore, developing effective RUL prediction methods using Few-Shot Learning (FSL) has become a new research focus. The LSSVM is one of the most commonly used FSL models owing to its outstanding performance in multidimensional nonlinear mapping, small sample processing and model generalization [17]. In order to define the time-to-start point of RUL prediction, Islam *et al*. [18] applied the LSSVM for anomaly detection. Zhang *et al*. [19] used the LSSVM model to extract the degraded nonlinear features of power electronic devices. Huang *et al*. [20] used the LSSVM model optimized by Particle Swarm Optimization (PSO) to predict the life of fuel cells. Thus, the LSSVM model is chosen in this paper.

3 Data preprocessing

In this part, we introduce the data preprocessing methods used in this study, including the data denoising, data normalization and feature engineering.

3.1 DB4 denoising

Actual data are prone to various types of noise contamination, which is detrimental to modeling. Thus, the Daubechies wavelet with four vanishing moments (DB4) denoising is introduced for data filtering due to its advantages, such as multiple vanishing moments, good regularity and approximate symmetry. The wavelet analysis theory indicates that any signal $Z(t)$ can be approximated using a series of orthogonal basis functions [21], that is

$$Z(t) = \sum_{j=-\infty}^{\infty} \alpha_j^I f_{I,j}(t) + \sum_{i=1}^{I} \sum_{k=-\infty}^{\infty} \beta_j^i g_{i,j}(t), \quad (1)$$

where i denotes the ith decomposition level, and j is the location of a basis function, $f_{I,j}(t)$ and $g_{i,j}(t)$ are the DB4 scaling functions at level I and the DB4 wavelet function at level i with respect to location j, respectively. α_j^I and β_j^i are the corresponding coefficients, $i = 1, 2, ..., I$.

In DB4 denoising, the components with small wavelet coefficients are discarded, while the components with large wavelet coefficients are retained. Finally, the denoised signal $\hat{Z}(t)$ is acquired by applying the inverse wavelet transform. We obtain the denoised dataset by applying DB4 denoising for every signal (i.e., feature).

3.2 Data normalization

To eliminate the influence of the index scale on modeling, the uniform extreme deviation processing method is utilized to standardize the dataset. Its calculation formula is

$$\tilde{Z}_i(t) = \frac{\hat{Z}_i(t) - \hat{Z}_i^{min}}{\hat{Z}_i^{max} - \hat{Z}_i^{min}}, \quad (2)$$

where $\tilde{Z}_i(t)$ is the normalized data of feature i, $\hat{Z}_i(t)$ represents the denoised data of the i-th feature $Z_i(t)$, $\hat{Z}_i^{min}$ and $\hat{Z}_i^{max}$ represent the maximum and minimum values of $\hat{Z}_i(t)$, respectively. Obviously, $\tilde{Z}_i(t) \in [0, 1]$, which helps to eliminate the influence of scale differences among features on modeling.

4 Research methodology

4.1 KPCA

As is well known, the quality and number of features directly determine the performance and complexity of the model. Therefore, feature engineering is highly necessary before modeling. KPCA is a nonlinear extension form of PCA [22] and is one of the most commonly used methods to accomplish nonlinear dimensionality reduction in feature engineering. In this paper, KPCA is used owing to its ability to capture nonlinear features. The main principle of KPCA is as follows [23].

Let $\{X_k^{tr}, Y_k^{tr}\}_{k=1}^{K_{tr}}$ denote the training dataset including K_{tr} samples, where $X_k^{tr} \in \mathbb{R}^{1\times n}$ is the k-th sample with n features, and Y_k^{tr} is the corresponding scalar output. According to the KPCA theory, the kernel matrix Ψ_{tr} can be calculated as:

$$\Psi_{tr}[i, j] = \Phi(X_i^{tr}, X_j^{tr}), i, j = 1, 2, ..., K_{tr}, \quad (3)$$

where $\Psi_{tr}[i, j]$ is the element in i-th row and j-th column of Ψ_{tr}, $\Phi(\cdot)$ is kernel function. Typically, $\Phi(\cdot)$ is chosen as the Gaussian kernel function [17], that is

$$\Phi(x_1, x_2) = e^{-\frac{||x_1 - x_2||^2}{2\sigma^2}}, \quad (4)$$

where $||\cdot||$ represents the Euclidean norm, and $\sigma > 0$ is the kernel bandwidth. The kernel matrix must be centralized. Thus, Ψ_{tr} is centralized as:

$$\tilde{\Psi}_{tr} = \left(I_{K_{tr}} - \frac{1_{K_{tr}}}{K_{tr}}\right) \Psi_{tr} \left(I_{K_{tr}} - \frac{1_{K_{tr}}}{K_{tr}}\right)^T, \quad (5)$$

where $I_{K_{tr}}$ is a K_{tr}-dimensional identity matrix, and $1_{K_{tr}}$ is a K_{tr}-dimensional matrix with all elements equal to 1.

Then, the i-th eigenvalue λ_i and eigenvector α_i of $\tilde{\Psi}_{tr}$ are obtained by solving the following equation.

$$\tilde{\Psi}_{tr}\alpha_i = \lambda_i \tilde{\Psi}_{tr}, \tag{6}$$

Next, let the eigenvalues satisfy $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_{K_{tr}}$, with the corresponding eigenvectors being α_1, α_2, ..., $\alpha_{K_{tr}}$. The cumulative contribution rate τ_k is then defined as:

$$\tau_k = \sum_{i=1}^{k} \lambda_i / \sum_{i=1}^{K_{tr}} \lambda_i, k = 1, 2, ..., K_{tr}. \tag{7}$$

Let τ_0 represent the threshold value of cumulative contribution rate. Assuming $\tau_{K-1} < \tau_0$ and $\tau_K \geq \tau_0$ $(2 < K \leq K_{tr})$, we obtain the feature matrix $\mathcal{A} = [\alpha_1, \alpha_2, ..., \alpha_K]$. Finally, the new dataset can be represented as:

$$X_{new}^{tr} = \tilde{\Psi}_{tr}\mathcal{A}, \tag{8}$$

In addition, considering a new dataset $\{X_k^{te}\}_{k=1}^{K_{te}}$, the joint kernel matrix Ψ_u can be computed as:

$$\Psi_u[i, j] = \Phi(X_i^{te}, X_j^{tr}), \tag{9}$$

where $\Psi_u[i, j]$ is the element in the i-th row and j-th column of Ψ_u, with $i = 1, 2, ..., K_{te}$ and $j = 1, 2, ..., K_{tr}$. The new kernel matrix Ψ_{te} can be computed as:

$$\Psi_{te} = \left(\Psi_u - \frac{1_{K_{te}K_{tr}}\Psi_{tr}}{K_{tr}}\right)\left(I_{K_{tr}} - \frac{1_{K_{tr}}}{K_{tr}}\right), \tag{10}$$

where $1_{K_{te}K_{tr}}$ denotes the dimensional matrix with the dimensions $K_{te} \times K_{tr}$ and all elements equal to 1. For the new dataset, its projection into the K-dimensional subspace is formulated as:

$$X_{new}^{te} = \Psi_{te}\mathcal{A}, \tag{11}$$

Remark 1 *: It is well-known that selecting an appropriate kernel function is crucial for KPCA. Since the Gaussian kernel function is widely used, we adopt it in this paper.*

4.2 LSSVM model

The LSSVM, proposed by Suykens *et al.* [24], is an improved version of the SVM. It replaces the inequality constraints of the SVM with equality constraints and can be solved as a system of linear equations. Due to its improved computational speed and reduced computational complexity [25], it is widely used in small-sample regression tasks. The description of the LSSVM model is as follows [26].

$$f(x) = w^T\varphi(x), +b \tag{12}$$

where w is the optimizable weight vector, b is the adjustable bias, and $\varphi(\cdot)$ is a nonlinear mapping function.

Then, based on the structural risk minimization criterion, its model parameters are obtained by optimizing the following objective function.

$$\begin{cases} \min\limits_{w,b,\xi_i} J(w, \xi_i) = \frac{1}{2}\left(\|w\|^2 + \gamma\sum_{i=1}^{K_{tr}}\xi_i^2\right) \\ \\ \text{s.t.} Y_i^{tr} = w \cdot \varphi(X_i^{tr}) + b + \xi_i, \end{cases} \tag{13}$$

where γ is the penalty factor (or regularization parameter), ξ_i is the relaxation (error) factor for ith sample.

For the optimization problem with equality constraints, the Lagrange equation V can be built

$$V(w, b, \xi, \eta) = J(w, \xi_i) - \sum_{i=1}^{K_{tr}} \eta_i \left[Y_i^{tr} - \tilde{Y}_i^{tr} - \xi_i\right] \tag{14}$$

where $\tilde{Y}_i^{tr} = w \cdot \varphi(X_i^{tr}) + b$, $\xi = [\xi_1, \xi_2, ..., \xi_{K_{tr}}]^T$, and $\eta = [\eta_1, \eta_2, ..., \eta_{K_{tr}}]^T$ is the Lagrange multiplier. According to the Karush-Kuhn-Tucker (KKT) conditions, calculate the partial derivatives of w, b, ξ, η in equation (14) respectively and set them to zero, that is

$$\frac{\partial V}{\partial w} = 0, \frac{\partial V}{\partial b} = 0, \frac{\partial V}{\partial \xi} = 0, \frac{\partial V}{\partial \eta} = 0, \tag{15}$$

Next, after eliminating the parameters w and ξ, we have

$$\begin{bmatrix} 0 \\ Y^{tr} \end{bmatrix} = \begin{bmatrix} 0 & 1_{1\times K_{tr}} \\ 1_{K_{tr}}^T & \mathcal{K} + \gamma^{-1}I_{K_{tr}} \end{bmatrix} \begin{bmatrix} b \\ \eta \end{bmatrix}, \tag{16}$$

where $1_{1\times K_{tr}}$ is a K_{tr}-dimensional row vector with all elements equal to 1, $\mathcal{K}$ is the kernel matrix of the LSSVM model with elements $\mathcal{K}[i, j] = \varphi(X_i^{tr}, X_j^{tr}), i, j = 1, 2, ..., K_{tr}$. Since the radial basis function (RBF) is the most commonly used nonlinear kernel function, it is chosen in our study. The RBF is given by

$$\varphi(X_i^{tr}, X_j^{tr}) = e^{-\frac{\|X_i^{tr} - X_j^{tr}\|^2}{\kappa^2}} \tag{17}$$

where κ is the kernel function parameter.

Finally, the LSSVM output is obtained as follows:

$$f(x) = \sum_{i=1}^{K_{tr}} \eta_i \varphi(x, X_i^{tr}), \quad (18)$$

As there are two hyperparameters in LSSVM (γ and κ) to be optimized, the TSO algorithm is employed for this purpose.

4.3 TSO algorithm

TSO is a swarm-based metaheuristic algorithm inspired by the cooperative foraging behavior of tuna swarm, i.e., spiral foraging and parabolic foraging [27]. It has been widely employed in many fields, such as power grid optimization operation [28] and risk assessment optimization for hydrogen energy storage [29]. The mathematical model of TSO is as follows.

- **Initialization**: TSO generates initial populations uniformly at random in the search space, that is

$$x_i^0 = r_0(x_{ub} - x_{lb}) + x_{lb}, i = 1, 2, 3, ..., N_p, \quad (19)$$

where x_i^0 is the i-th initial individual, r_0 is a random number uniformly distributed between 0 and 1, x_{ub} and x_{lb} are the upper and lower bounds of the search space, and N_p is the number of individuals in the tuna group.

- **Spiral foraging**: The tuna group chases their prey by forming a tight spiral formation around each other. This spiral foraging strategy can be mathematically described as

$$x_i^{k+1} = c_1(x_{best}^k + \beta|x_{best}^k - x_i^k|) + c_2 x_{i-m}^k, \quad (20)$$

where x_i^{k+1} is the i-th individual at $k+1$ iteration, x_{best}^k is the current best individual, c_1 and c_2 are weight coefficients, β is the distance scaling factor between a general individual and the optimal individual, k denotes the number of current iteration, and m is a binary variable such that $m = 0$ if $i = 1$ or $m = 1$ otherwise. c_1, c_2 and β are defined as follows.

$$c_1 = c + (1 - c)\frac{k}{k_{max}}, \quad (21)$$

$$c_2 = (1 - c)(1 - \frac{k}{k_{max}}), \quad (22)$$

$$\beta = e^{be^{3\cos((k_{max}+1/k)\pi)}} \cos(2\pi b), \quad (23)$$

where k_{max} is the maximum number of iterations, b is a uniformly distributed random number between 0and 1. c is a constant with a default value of 0.7.

To avoid over-reliance on the optimal individual and improve the global exploration ability of TSO, a random reference point is introduced for spiral search within the search space. The mechanism is described as follows.

$$x_i^{k+1} = c_1(x_{round}^k + \beta|x_{round}^k - x_i^k|) + c_2 x_{i-m}^k, \quad (24)$$

where x_{round}^k is a randomly generated coordinate in the search space. In TSO, individuals randomly choose either equation (20) or equation (24) to update its position. Thus, we have

$$x_i^{k+1} = c_1(x_T^k + \beta|x_T^k - x_i^k|) + c_2 x_{i-m}^k, \quad (25)$$

where $x_T^k = x_{best}^k$ if $\zeta_1 \leq k/k_{max}$ or $x_T^k = x_{round}^k$ otherwise (where ζ_1 is a random number uniformly distributed between 0 and 1).

- **Parabolic foraging**: In addition to the spiral foraging, tuna swarms also engage in parabolic foraging. This process is described as follows.

$$x_i^{k+1} = \begin{cases} x_{best}^k + r_1 e_i^k + r_2 p^2 e_i^k, \zeta_2 < 0.5, \\ r_2 p^2 x_i^k, \quad \text{otherwise.} \end{cases}, \quad (26)$$

where $p = (1 - k/k_{max})^{k/k_{max}}$, r_1 is a uniformly distributed random number between 0 and 1, and r_2 is a random variable that takes the value of either 1 or -1.

In the TSO, each individual randomly chooses among initialization (19), spiral foraging (25) or parabolic foraging (26) in each iteration. Overall, the complete mathematical model of the TSO is as follows.

$$x_i^{k+1} = \begin{cases} r_0(x_{ub} - x_{lb}) + x_{lb}, & \text{st1}, \\ x_{best}^k + r_1 e_i^k + r_2 p^2 e_i^k, & \text{st2}, \\ r_2 p^2 x_i^k, & \text{st3}, \\ c_1(x_T^k + \beta|x_T^k - x_i^k|) + c_2 x_{i-m}^k, & \text{st4}. \end{cases}, \quad (27)$$

where st1 represents $\zeta_3 < \zeta$, st2 represents $\zeta_2 < 0.5$ & $\zeta_3 \geq \zeta$ & $\zeta_4 < 0.5$, st3 represents $\zeta_2 \geq 0.5$ & $\zeta_3 \geq \zeta$ & $\zeta_4 < 0.5$ and st4 represents $\zeta_3 \geq \zeta$ & $\zeta_4 \geq 0.5$. Here, ζ_3 and ζ_4 are random numbers uniformly distributed on the interval [0, 1], while ζ is the probability for selecting an initialization, with a default values as 0.05.

In the paper, TSO is employed to optimize hyperparameters γ and κ in the LSSVM model. The mean squared error (MSE) is selected as the

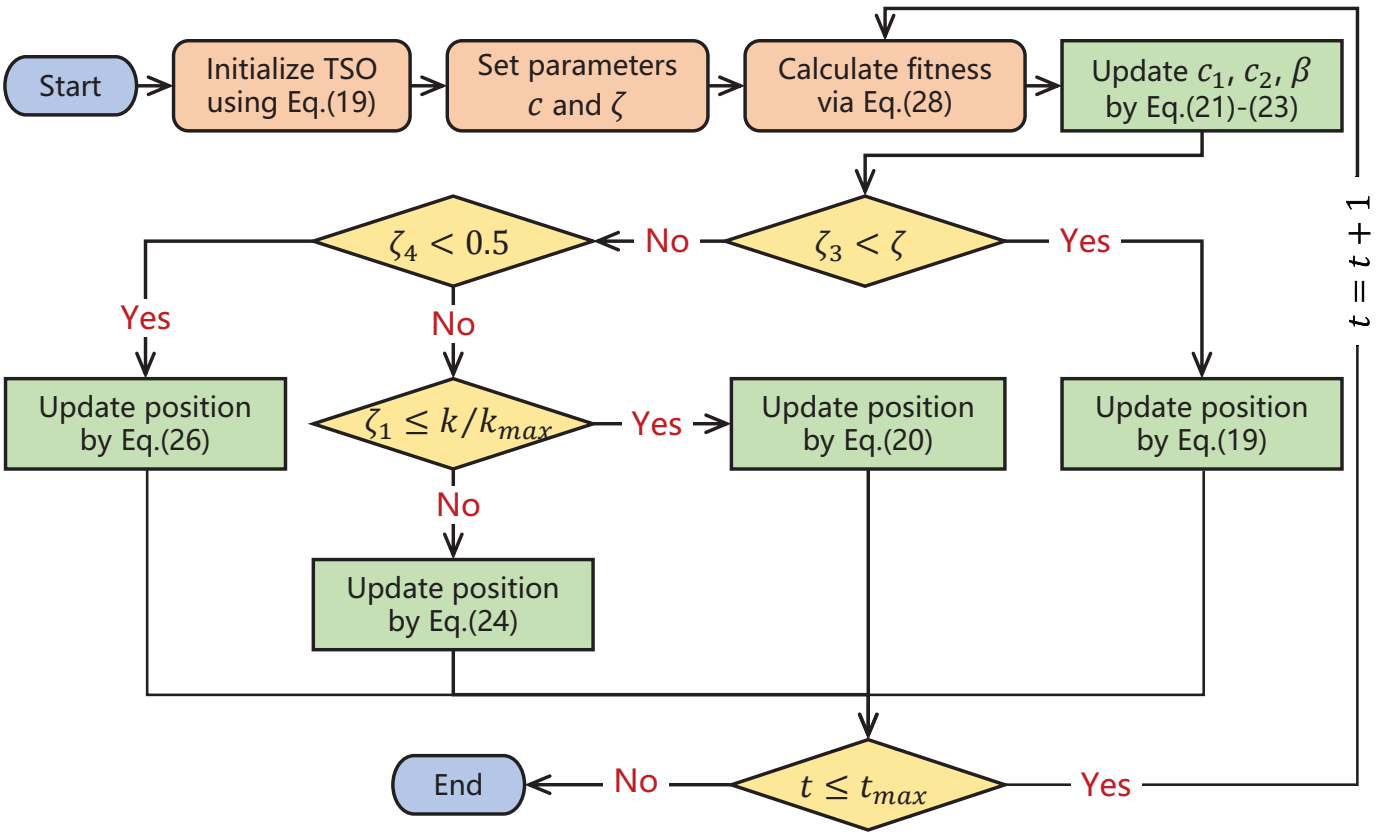

Figure 1. Flowchart of TSO.

objective function for the optimization problem, that is

$$\begin{cases} \min\limits_{\gamma,\kappa} J(\gamma,\kappa) = \frac{1}{K_{tr}} \sum_{i=1}^{K_{tr}} (Y_i^{tr} - \hat{Y}_i^{tr})^2 + \\ \qquad \epsilon^2(\gamma^2 + \kappa^2), \\ \text{s.t.} \gamma \in (0, \gamma_{ub}], \kappa \in (0, \kappa_{ub}], \end{cases} \tag{28}$$

where $\hat{Y}_i^{tr}$ is the i-th sample output of the LSSVM, which can be calculated by equation (18). $\epsilon = 1/\max(\gamma_{ub}, \kappa_{ub})$ is the regularization coefficient.

All individuals of TSO are iteratively updated by equation (27). The optimal individual (γ^*, κ^*) and the corresponding fitness value are returned when $k > k_{max}$. The detailed process of TSO is shown in Figure 1.

4.4 RUL prediction based on DKT-LSSVM

Firstly, we use the DB4 denoising method to eliminate noise in the original data. Then, KPCA is utilized to extract the principal components of the denoised data. Finally, these principal components are used as input features for the LSSVM model. Moreover, the TSO algorithm is used to optimize the hyperparameters of the LSSVM model owing to its strong global search capability. The process of RUL prediction based on DKT-LSSVM is illustrated in Figure 2. The implementation steps of the proposed algorithm are summarized in Algorithm 1.

In Figure 2, $D_{tr} = [X_{tr}, Y_{tr}]$, $D_{te} = [X_{te}, Y_{te}]$, $X_{tr} = \{X_k^{tr}\}_{k=1}^{K_{tr}}$, $Y_{tr} = \{Y_k^{tr}\}_{k=1}^{K_{tr}}$, $X_{te} = \{X_k^{te}\}_{k=1}^{K_{te}}$, $Y_{te} = \{Y_k^{te}\}_{k=1}^{K_{te}}$, and $\hat{Y}$ is the predicted service life.

5 Case

5.1 Data sources and preprocessing

The study focuses on the injection lance used in copper top-blown smelting. We collected some actual production data from a certain smelter, including Actual production data, comprising 1613 samples and 33 features, were collected from a smelter. Part of the dataset is presented in Table 1.

The variables are defined as follows: x_1 is the O_2 consumption (Nm^3), x_2 is the air consumption of the lance (Nm^3), and x_3 is the average temperature. Additionally, x_{33} is the air consumption of the burner, and y is the service life.

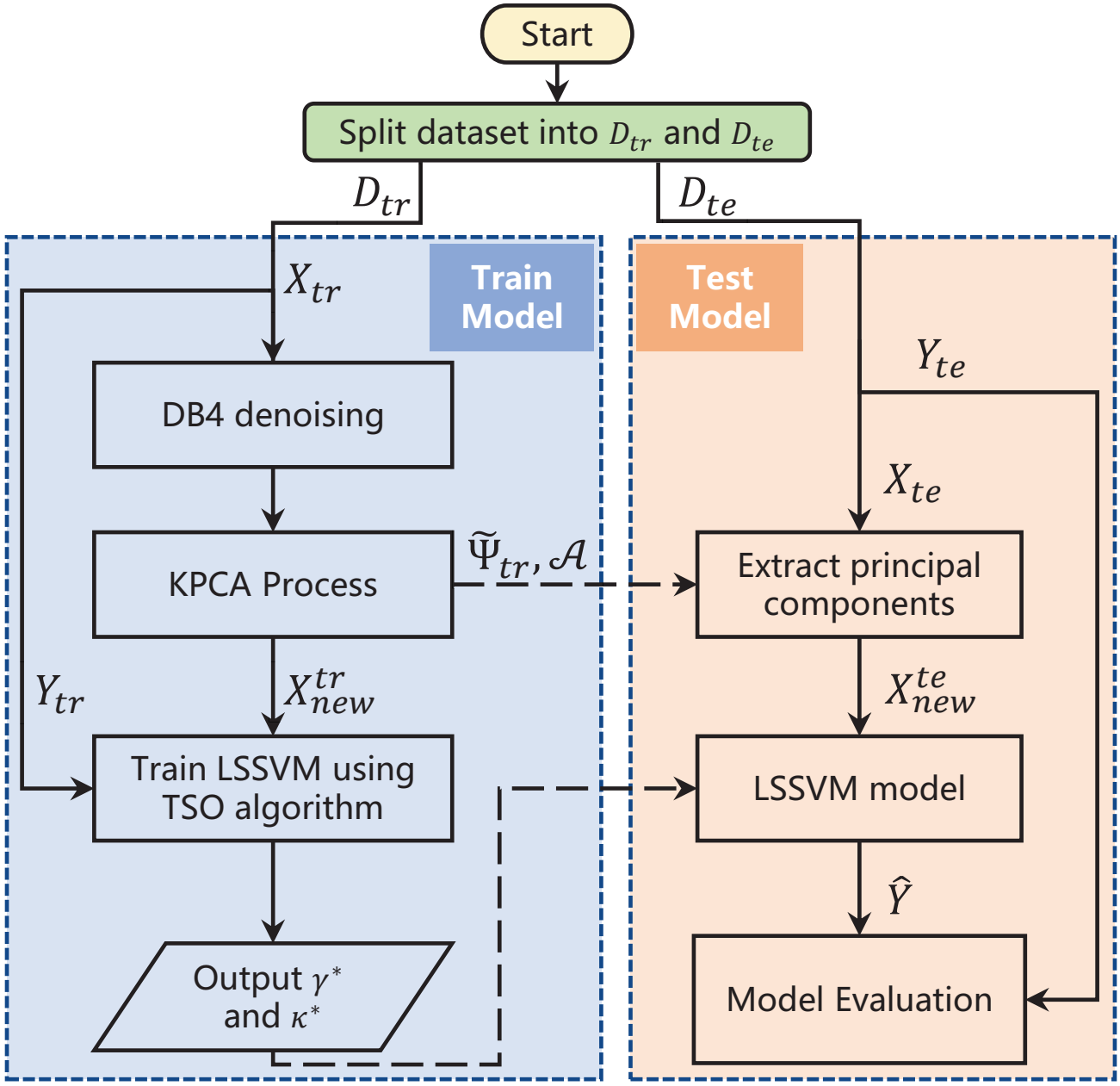

Figure 2. Flowchart of the proposed prediction model.

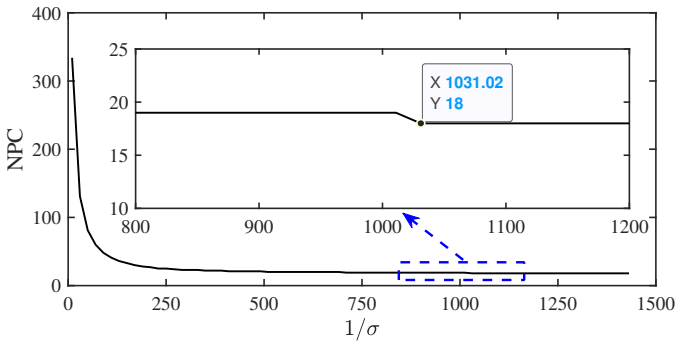

Figure 3. Fitness curve of the iterative process for optimizing σ. NPC is the number of principal components.

For each feature, the DB4 denoising is applied to remove noise from the original data. Each feature is then normalized using the max-min normalization method (2). In the experiment, we used Python 3.11.7 for modeling on a computer equipped with an Intel(R) Core(TM) i9-14900K 3.20 GHz processor and 128 GB of RAM.

5.2 Model building

In the KPCA, we set the threshold of the cumulative contribution rate $\tau_0 = 0.90$ [30]. The kernel parameter of KPCA σ is optimized by the grid search in the range of [10, 100]. The training process is shown in Figure 3. Therefore, we set $\sigma^* = 53.00$. Then, the contribution rates and cumulative contribution rates of the principal

Algorithm 1: RUL prediction algorithm based on KPCA-TSO-LSSVM.

Collect the operational data of the injection lance. Then, the key steps are as follows.

- **Train model**
  1. **DB4 denoising**
     Remove noise from the original data using the DB4 denoising principle.
  2. **Normalization**
     Normalize each feature via equation (2).
  3. **Divide dataset**
     Normalized dataset is randomly divided into a training dataset $\{X_k^{tr}, Y_k^{tr}\}_{k=1}^{K_{tr}}$ (80% of the dataset) and a test dataset $\{X_k^{te}, Y_k^{te}\}_{k=1}^{K_{te}}$ (20% of the dataset).
  4. **KPCA**
     Calculate the centralized kernel matrix $\tilde{\Psi}_{tr}$, the feature matrix $\mathcal{A}$ and Ψ_{te} using equation (5), equations (6 - 7) and equation (10), respectively. Then, obtain the new datasets X_{new}^{tr} using equation (8) and X_{new}^{te} using equation (11).
  5. **TSO optimizes LSSVM**
     Set the population size N_p and the maximum number of iterations k_{max}. Then, update each TSO individual using equation (27) and evaluate its fitness using equation (28). When $k > k_{max}$, return the optimal individual (γ^*, κ^*).
- **Test model** The test dataset $\{X_k^{te}, Y_k^{te}\}_{k=1}^{K_{te}}$ is input into the LSSVM model to obtain the RUL prediction of the injection lance. Then, the model performance is further evaluated by some metrics.

components are shown in Figure 4.

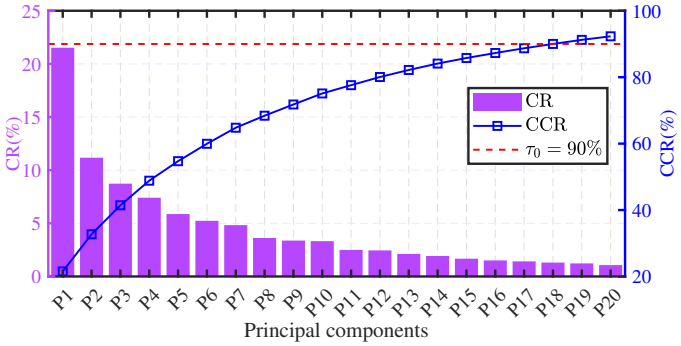

Figure 4. Contribution rates and cumulative contribution rates of the principal components. CR is contribution rates and CCR is cumulative contribution rates.

As shown in Figure 4, the cumulative contribution of the first 18 principal components (P1 to P18) is greater than 90%. Thus, these principal components are selected to as the model input.

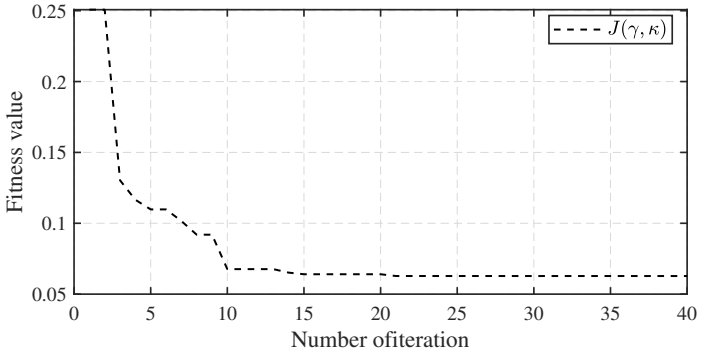

Figure 5. Fitness curve of the training process.

In addition, the parameters of the TSO algorithm are listed in Table 2.

Table 2. Parameters of the TSO algorithm.

Parameter	Symbol	Value
Initial population	N_p	20
Maximum iteration	k_{max}	40
Upper boundary	$[\gamma_{ub}, \kappa_{ub}]$	[500, 500]
Constant	c	0.7
Tuna mutation probability	ζ	0.05
Kernel function of LSSVM	$\varphi(\cdot)$	RBF

Based on the parameter configuration in Table 2, we used the TSO to train the proposed Algorithm 1. The final results were $\gamma^* = 65.9355$ and $\kappa^* = 45.1149$.

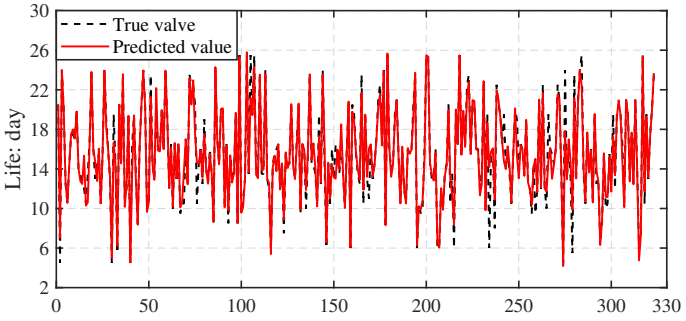

Figure 6. True and predicted life curves of the injection lance in the test dataset.

5.3 Evaluation metrics

To objectively evaluate the predictive performance of the model, the Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and coefficient of determination (R^2) are adopted as evaluation metrics, with their

Table 1. Part of the dataset information.

ID	x_1	x_2	x_3	...	x_{33}	y
1	110296	217680	1146.25	...	33526	12.5
2	96586	205167	1140.25	...	20082	20.5
3	110926	213310	1155.00	...	23584	20.5
4	113175	217757	1173.00	...	25516	20.5
...	...	...	...	...	...	...
1613	119720	77753	1146.75	...	17147	6.0

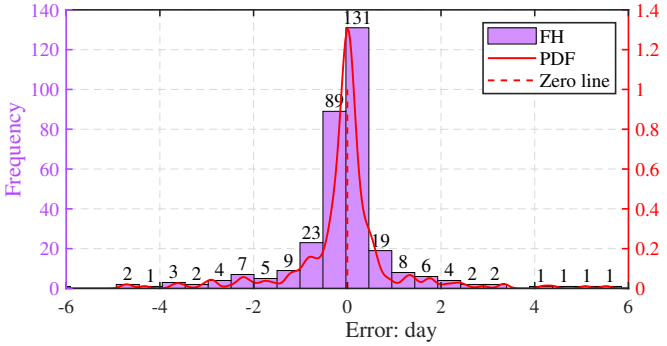

Figure 7. Histograms of the data distribution in the test dataset.

definitions as follows respectively.

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(y_i - \hat{y}_i)^2}, \tag{29}$$

$$\mathrm{MAE} = \frac{1}{N}\sum_{n=1}^{N}|y_i - \hat{y}_i|, \tag{30}$$

$$\mathrm{R}^2 = 1 - \sum_{n=1}^{N}(y_i - \hat{y}_i)^2 / \sum_{n=1}^{N}(y_i - \bar{y})^2, \tag{31}$$

where y_i is the true value, $\hat{y}_i$ is the predicted value, N is the number of samples, $\bar{y} = \frac{1}{N}\sum_{n=1}^{N} y_i$ is the mean value. Obviously, the closer RMSE and MAE are to 0 and the closer R^2 is to 1, the better the model performance.

5.4 Results and analysis

In this part, we demonstrate the performance of the proposed algorithm and conducted comparative experiments. The true and predicted life curves of the injection lance in the test dataset are shown in Figure 6. Furthermore, the histogram and probability density function of the prediction errors in test datasets are shown in Figure 7. The median of the error distribution curves for the test dataset is very close to 0. Therefore, as it can be seen from Figures 6 and 7, the proposed algorithm can successfully capture the nonlinear relationship between the RUL of injection lance and the influencing factors.

In addition, during the lifecycle of an injection lance, the prediction results of its RUL are shown in Figure 8. The Figure 8 (left) shows that the proposed algorithm provides relatively reliable RUL prediction for the injection lance, with the maximum absolute error not exceeding 0.4 days.

Finally, to verify the superiority of the proposed algorithm, comparative experimental results are presented. For these experiments, we selected the following methods for comparison: Extreme Learning Machine (ELM), Back Propagation Neural Network (BPNN), Recurrent Neural Network (RNN), Random Forest (RF), Least Squares Support Vector Machine (LSSVM), Support Vector Machine (SVM) and KSSVM. Note that KSSVM refers to an SVM optimized by the Sparrow Search Algorithm (SSA) with KPCA for feature extraction. The evaluation metrics of different methods are shown in Table 3.

Table 3. The evaluation metrics of different methods.

Methods	RMSE(day)	MAE(day)	R^2
Our method	**1.2274**	**0.6623**	**0.9308**
KSSVM	2.32	1.56	0.78
SVM	2.65	1.94	0.68
LSSVM	2.72	1.97	0.68
RF	2.86	2.07	0.66
RNN	3.55	2.73	0.43
BPNN	3.86	2.92	0.42
ELM	4.07	3.19	0.27

The results show that our method outperforms the other models in terms of the service life prediction of the injection lance. Analysis of Table 3 shows that:

1. In the task of the service life prediction for the injection lance, the symbolism-based models (such as DKT-LSSVM, KSSVM, SVM, LSSVM and RF) generally perform better than connectionism-based models (e.g., BPNN, RNN and ELM). This indicates that symbolism-based models are more suitable for this task.

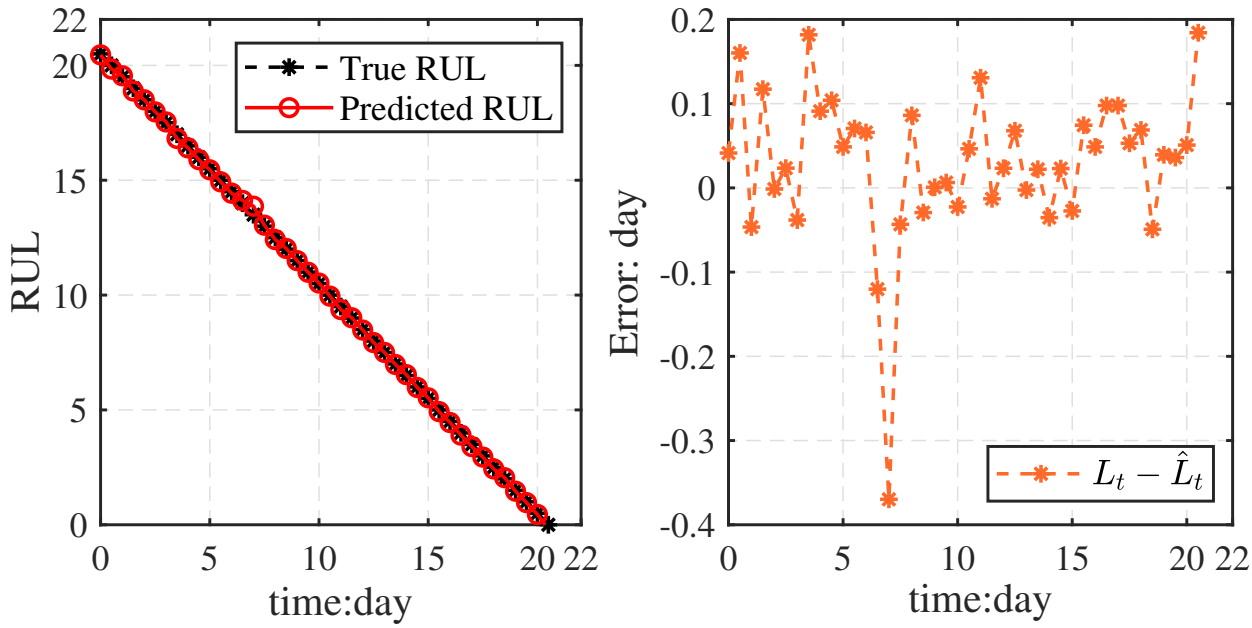

Figure 8. The prediction results of RUL for a injection lance during its lifecycle. True and predicted RUL of a injection lance (left) and predicted error of its RUL (right). L_t is true RUL and $\hat{L}_t$ is predicted RUL.

2. KPCA can effectively extract the key information from the nonlinear influencing factors and remove redundant information. Therefore, our method and KSSVM perform better than the others.
3. Since the LSSVM is an improvement over SVM, it offers better fitting capability. However, compared with the KSSVM, our algorithm performs better.

6 Conclusions

This paper focuses on the RUL prediction problem of the injection lance in the copper top-blown smelting and proposes a DKT-LSSVM method for the soft measurement of RUL of the injection lance. The results demonstrate that the proposed method can reliably predict the RUL of the injection lance. In comparative experiments, the proposed method achieves the best evaluation metrics (RMSE=1.2274 day, MAE=0.6623 day and R^2=0.9308), which further verifies the effectiveness and superiority of the proposed method. Thus, the proposed method can be applied to the RUL prediction of the injection lance in the copper top-blown smelting. It not only provides a reference for timely replacing the injection lance, but also enhances the automation level of the copper top-blown smelting. Additionally, the proposed method can be further extended to predict the life of other equipment, such as the P-S converter and reduction furnace in copper bath smelting.

Data Availability Statement

Data will be made available on request.

Funding

This work was supported in part by the Yunnan Major Scientific and Technological Projects, China under Grant 202302AD080005; in part by the Yunnan Fundamental Research Projects under Grant 202301AT070401.

Conflicts of Interest

Qiang Peng is an employee of Huize Smelting Branch, Yunnan Chihong Zinc and Germanium Co., Ltd., Huize 654200, China. The authors declare no conflicts of interest.

Ethical Approval and Consent to Participate

Not applicable.

References

[1] Zhao, H. L., Yin, P., Zhang, L. F., & Wang, S. (2016). Water model experiments of multiphase mixing in the top-blown smelting process of copper concentrate. *International Journal of Minerals, Metallurgy, and Materials, 23*(12), 1369-1376. [CrossRef]

[2] Floyd, J. M. (1996). The third decade of top submerged lance technology. In *The Howard Worner International Symposium on Injection in Pyrometallurgy* (pp. 417-429).

[3] Wan, Z., Yang, S., Kong, D., Li, D., Hu, J., & Wang, H. (2024). Numerical investigation of sinusoidal pulsating gas intake to intensify the

gas-slag momentum transfer in the top-blown smelting furnace. *International Journal of Minerals, Metallurgy and Materials, 31*(2), 301-314. [CrossRef]

[4] Zhao, H., Lu, T., Liu, F., Yin, P., & Wang, S. (2019). Computational fluid dynamics study on a top-blown smelting process with lance failure in an Isa furnace. *JOM, 71*(5), 1643-1649. [CrossRef]

[5] Yu, B., Guo, H., & Shi, J. (2025). Remaining useful life prediction based on hybrid CNN-BiLSTM model with dual attention mechanism. *International Journal of Electrical Power & Energy Systems, 172*, 111152. [CrossRef]

[6] Li, J., Liu, Y., & Li, Q. (2022). Intelligent fault diagnosis of rolling bearings under imbalanced data conditions using attention-based deep learning method. *Measurement, 189*, 110500. [CrossRef]

[7] Jia, F., Lei, Y., Lin, J., Zhou, X., & Lu, N. (2016). Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. *Mechanical systems and signal processing, 72*, 303-315. [CrossRef]

[8] Chen, J., Huang, R., Chen, Z., Mao, W., & Li, W. (2023). Transfer learning algorithms for bearing remaining useful life prediction: A comprehensive review from an industrial application perspective. *Mechanical Systems and Signal Processing, 193*, 110239. [CrossRef]

[9] Mathew, V., Toby, T., Singh, V., Rao, B. M., & Kumar, M. G. (2017, December). Prediction of Remaining Useful Lifetime (RUL) of turbofan engine using machine learning. In *2017 IEEE international conference on circuits and systems (ICCS)* (pp. 306-311). IEEE. [CrossRef]

[10] Mo, R., Zhou, H., Yin, H., & Si, X. (2025). A survey on few-shot learning for remaining useful life prediction. *Reliability Engineering & System Safety*, 110850. [CrossRef]

[11] Song, Y., Gao, S., Li, Y., Jia, L., Li, Q., & Pang, F. (2020). Distributed attention-based temporal convolutional network for remaining useful life prediction. *IEEE Internet of Things Journal, 8*(12), 9594-9602. [CrossRef]

[12] Zhang, X., Dong, Y., Wen, L., Lu, F., & Li, W. (2019, August). Remaining useful life estimation based on a new convolutional and recurrent neural network. In *2019 ieee 15th international conference on automation science and engineering (case)* (pp. 317-322). IEEE. [CrossRef]

[13] Zheng, S., Ristovski, K., Farahat, A., & Gupta, C. (2017, June). Long short-term memory network for remaining useful life estimation. In *2017 IEEE international conference on prognostics and health management (ICPHM)* (pp. 88-95). IEEE. [CrossRef]

[14] Zhang, Y., Xiong, R., He, H., & Pecht, M. G. (2018). Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries. *IEEE Transactions on Vehicular Technology, 67*(7), 5695-5705. [CrossRef]

[15] Wang, J., Wen, G., Yang, S., & Liu, Y. (2018, October). Remaining useful life estimation in prognostics using deep bidirectional LSTM neural network. In *2018 Prognostics and System Health Management Conference (PHM-Chongqing)* (pp. 1037-1042). IEEE. [CrossRef]

[16] Zhong, S. S., Fu, S., & Lin, L. (2019). A novel gas turbine fault diagnosis method based on transfer learning with CNN. *Measurement, 137*, 435-453. [CrossRef]

[17] Zhang, K., Zhang, K., & Bao, R. (2023). Prediction of gas explosion pressures: A machine learning algorithm based on KPCA and an optimized LSSVM. *Journal of Loss Prevention in the Process Industries, 83*, 105082. [CrossRef]

[18] Islam, M. M., Prosvirin, A. E., & Kim, J. M. (2021). Data-driven prognostic scheme for rolling-element bearings using a new health index and variants of least-square support vector machines. *Mechanical Systems and Signal Processing, 160*, 107853. [CrossRef]

[19] Zhang, J., Hu, J., You, H., Jia, R., Wang, X., & Zhang, X. (2021). A remaining useful life prediction method of IGBT based on online status data. *Microelectronics Reliability, 121*, 114124. [CrossRef]

[20] Huang, W., Liu, M., Zhang, C., Niu, T., Fu, Z., Ren, X., & Chin, C. S. (2024). Life prediction for proton exchange membrane fuel cell based on experimental results and combinatorial optimization algorithm. *International Journal of Hydrogen Energy, 79*, 364-376. [CrossRef]

[21] Mallat, S. G. (2002). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE transactions on pattern analysis and machine intelligence, 11*(7), 674-693. [CrossRef]

[22] Liu, M., Yao, X., Zhang, J., Chen, W., Jing, X., & Wang, K. (2020). Multi-sensor data fusion for remaining useful life prediction of machining tools by IABC-BPNN in dry milling operations. *Sensors, 20*(17), 4657. [CrossRef]

[23] Zhou, T., & Peng, Y. (2020). Kernel principal component analysis-based Gaussian process regression modelling for high-dimensional reliability analysis. *Computers & Structures, 241*, 106358. [CrossRef]

[24] Suykens, J. A., & Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural processing letters, 9*(3), 293-300. [CrossRef]

[25] Ansari, S., Safaei-Farouji, M., Atashrouz, S., Abedi, A., Hemmati-Sarapardeh, A., & Mohaddespour, A. (2022). Prediction of hydrogen solubility in aqueous solutions: Comparison of equations of state and advanced machine learning-metaheuristic approaches. *International Journal of Hydrogen Energy, 47*(89), 37724-37741. [CrossRef]

[26] Tian, Z. (2020). Short-term wind speed prediction based on LMD and improved FA optimized combined kernel function LSSVM. *Engineering Applications of*

Artificial Intelligence, 91, 103573. [CrossRef]

[27] Xie, L., Han, T., Zhou, H., Zhang, Z. R., Han, B., & Tang, A. (2021). Tuna swarm optimization: a novel swarm-based metaheuristic algorithm for global optimization. *Computational intelligence and Neuroscience, 2021*(1), 9210050. [CrossRef]

[28] Zhang, T., Wang, Q., Shu, Y., Xiao, W., & Ma, W. (2023). Remaining useful life prediction for rolling bearings with a novel entropy-based health indicator and improved particle filter algorithm. *IEEE Access, 11*, 3062-3079. [CrossRef]

[29] Liu, J., Song, Y., & Yu, X. (2024). Risk assessment study of hydrogen energy storage system based on KPCA-TSO-LSSVM. *International Journal of Hydrogen Energy, 79*, 931-942. [CrossRef]

[30] Kong, D., Chen, Y., & Li, N. (2018). Gaussian process regression for tool wear prediction. *Mechanical systems and signal processing, 104*, 556-574. [CrossRef]

Qiang Peng is a senior engineer and the head of Information and Automation of the Equipment and Energy Department of Huize Smelting Branch of Yunnan Chihong Zinc Germanium Co., Ltd. He has been committed to the field of non-ferrous metal smelting for a long time, especially the research and engineering practice of automation, informatization and intelligent technology of lead and zinc smelting process. His main research interests are in the areas of optimization of basic automation system of smelting process, industrial data acquisition and intelligent perception, construction of intelligent manufacturing infrastructure, and intrinsic safety and energy efficiency management of process industry. (Email: temp_chxz@163.com)

Gengen Li pursued the M.S. degree in Mechanical Engineering at the Faculty of Mechanical and Engineering, Kunming University of Science and Technology, Yunnan, China, from 2020 to 2022. In 2022, he obtained the qualification for a combined master's and doctoral program. Presently, he is pursuing the Ph.D. degree in mechanical manufacturing and automation at the same institution. His current research interests encompass multisource information fusion, optimal state estimation, and mechanism-enhanced data-driven modeling and optimization. (Email: lgg_root@163.com)

Chunxi Yang is a professor in the Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology. He received his BE degree in process control from Qingdao University of Science and Technology, Qingdao, China, in 1999, ME degree from Wuhan Institute of Technology, Wuhan, China, in 2005, and PhD degree from Huazhong University of Science and Technology, Wuhan, China, in 2009. His main research interests are in the areas of unmanned system design and control, distributed filtering and fusion, intelligent control system and industrial big data analyzation. (Email: ycx@kmust.edu.cn)

Mou Li received the B.E. degree in Intelligent Manufacturing Technology from Chengdu Technological University, Chengdu, China, in 2024. He is currently pursuing the M.E. degree in Intelligent Manufacturing Technology at Kunming University of Science and Technology, Kunming, China. His research interests include industrial big data analysis and intelligent manufacturing systems. (Email: 2047127804@qq.com)