Research on Engineering Structures & Materials
www.jresm.org
Research Group

Research Article

A hybrid deep learning-evolutionary optimization framework for intelligent prediction of calcium addition in molten steel refining processes

Yanan Liu [1,a], Juanjuan Li [1,b], Xiaoao Li [*,1,2,c], Shenghui Wang [1,d]

[1]*School of Intelligent Manufacturing, Xuchang Vocational Technical College, Xuchang, China*
[2]*Henan University of Technology Postdoctoral Mobile Station, Zhengzhou, China*

Article Info

Article History:

Received 20 Aug 2026
Accepted 09 Oct 2026

Keywords:

Calcium treatment prediction;
Multi-Modal data fusion;
CNN-BiLSTM-Attention;
Adaptive differential evolution;
Soft computing

Abstract

Effective prediction of calcium addition in molten steel treatment is important for inclusion modification and clean steel manufacturing. This study presents an application-oriented framework that integrates multimodal feature fusion, a CNN-BiLSTM-Attention prediction network, adaptive differential evolution, and SHapley Additive exPlanations (SHAP) for calcium-addition prediction. The fusion module uses modality-level attention to combine process parameters, chemical composition measurements, and sensor signals. The prediction network extracts local patterns, models bidirectional sequence dependencies, and weights informative sequence positions. An improved adaptive differential evolution algorithm with chaotic initialization and self-adaptive parameter control jointly searches network hyperparameters and input feature subsets. On the reported split of 2 846 industrial heats, the framework achieved an RMSE of 0.1038 kg/t and an R^2 of 0.9843 and produced the lowest reported error among the six compared baselines. The ablation comparisons indicate complementary contributions from the tested components, while SHAP analysis yields feature rankings that are broadly consistent with metallurgical knowledge. These findings support the feasibility of the integrated framework under the reported experimental configuration, although broader validation across steel grades, plants, and repeated random splits remains necessary.

© 2026 MIM Research Group. All rights reserved.

1. Introduction

The application of calcium is an essential step in metallurgy in the secondary refining of molten steel, which is mainly used to transform solid alumina inclusions to liquid calcium aluminates in order to avoid submergence nozzle plugging during continuous casting and enhance the quality of steel products [1]. As can be seen from Fig. 1, the usual calcium treatment procedure consists of adding the calcium-containing wire or alloy into the ladle furnace in the course of refinement when calcium interacts with oxide and sulfide inclusions under high-temperature and high-pressure conditions. Nonetheless, the low boiling point and the limited solubility of calcium in liquid steel cause its immediate vaporization after contacting with the molten steel, which makes the calcium yield unsteady and uncontrollable [2]. The exact amount of calcium which needs to be added is determined by many parameters, such as the initial oxygen and sulfur contents, aluminum content, molten steel temperature, addition rate, and slag composition [3]. The complicated nonlinear relations between all these parameters make conventional empirical approach to addition insufficient to achieve a stable inclusion modification process [4].

The principal reaction directions can be represented schematically as Al_2O_3(inclusion) + Ca → CaO–Al_2O_3(inclusion) and Ca + [S] → CaS(inclusion). These equations describe competing reaction

*Corresponding author: xiaoao_li@outlook.com
DOI: http://dx.doi.org/10.17515/resm2026-2227vk0820rs

pathways rather than equilibrium calculations. The suitable calcium-treatment window is therefore determined by the balance between oxide modification, sulfide formation, calcium vaporization, and mass transfer [1,4].

Calcium utilization is also governed by competing thermodynamic and kinetic pathways. Dissolved calcium can react with alumina inclusions to form progressively calcium-richer aluminates, including liquid calcium-aluminate phases within suitable composition and temperature ranges [1, 3, 4]. At the same time, sulfur consumes dissolved calcium through CaS formation, thereby competing with oxide modification. Temperature affects calcium solubility, vapor pressure, mass transfer, and reaction kinetics. Consequently, the effective calcium demand depends not only on the nominal inclusion burden but also on [Al], [T.O], [S], steel temperature, and slag conditions. During cored-wire feeding, these coupled reactions determine whether calcium promotes liquid inclusion formation, is lost by vaporization, or is consumed as CaS.

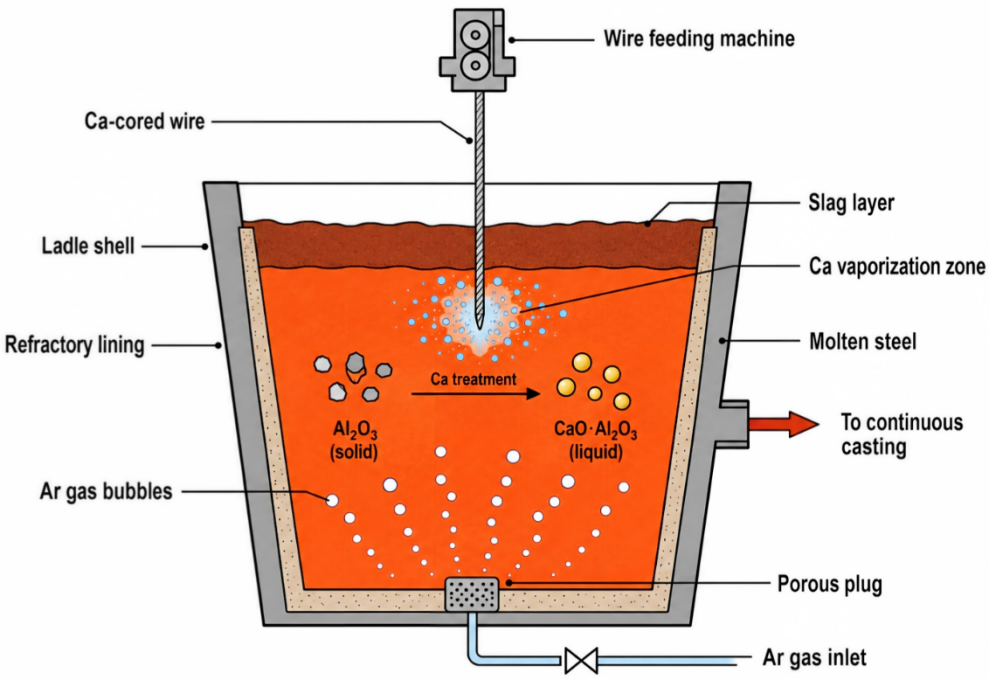


Fig. 1. Schematic diagram of the calcium treatment process during molten steel refining

Empirical formulas and thermodynamic theories have long constituted the main means to guide calcium addition; however, the traditional approach has relied too much on assumptions and operator adjustments, which makes them inapplicable in actual industrial settings where process variables change constantly [1]. In recent years, data-driven calcium addition approaches based on machine learning have been proposed to address these issues. Some classical machine learning algorithms like support vector regression, random forest, and backpropagation neural network have been employed to predict some steelmaking process variables like endpoint carbon and temperature, alloy yield, and molten iron quality [5, 6]. Although the data-driven approach outperforms the empirical approach in many aspects, it is prone to low representation capability when handling highly-dimensional and nonlinear process data with time dependence, and thus lacks prediction precision and adaptability to different grades of steels [7]. The neural network-based model was created specifically to make predictions about calcium yield during the calcium treatment process, suggesting that the data-driven calcium yield prediction is feasible, yet with the problem of low accuracy due to shallowness and single source of input [2].

Deep learning algorithms, especially recurrent neural networks, together with their gated counterparts, show better results in learning temporal dependencies from sequential datasets in industry. The long short-term memory networks, introduced to overcome the vanishing gradient problem in sequential learning [8], have already been used in molten steel temperature prediction, alloy element yield estimation, and silicon content prediction in blast furnace operations [9, 10]. Hybrid deep learning algorithms, such as those using convolutional neural networks and

bidirectional LSTM with attention mechanism, which are capable of simultaneous local feature extraction and bidirectional learning, have also achieved much higher prediction results in industrial process modeling, including rolling force prediction in cold rolling and blast furnace ironmaking heat load estimation [11, 12]. In addition, the self-attention mechanism developed within the context of the transformer architecture [13] helps to improve the focus on relevant parts of the input sequence in hybrid deep architectures and shows good results when combined with CNN-BiLSTM for multivariate time series prediction in soft computing [14]. Nonetheless, it is worth mentioning that no research on using these deep learning architectures for calcium treatment prediction exists, and in general, there is no hyperparameter optimization or multi-source data integration in deep learning models in metallurgy.

The results of deep learning models greatly depend on the choice of hyperparameters like the learning rate, number of hidden neurons, kernel size, and the dropout rate. Manual optimization of such parameters requires much time and is likely to produce non-optimal values. Bio-inspired algorithms can be employed for the automated search of the optimal configuration of hyperparameters. Differential evolution, being one of the most general population-based metaheuristics to solve global optimization problems with continuous variables [15] has been actively applied to various problems and considerably improved by methods that include adaptive tuning of parameters, development of effective mutations, and the use of differential evolution as a component of other search approaches [16]. Self-adaptive scaling factor and crossover rate were used to balance between exploration and exploitation [17], and chaotic map-based methods were employed for the population initialization to increase the diversity of solutions in the beginning of search [18]. A combination of particle swarm optimization and LSTM networks was also applied to the problem of predicting alloy yield in the process of steelmaking [10]. The majority of hybrid algorithms combining optimization and deep learning methods work on selecting a certain set of hyperparameters but do not consider the problem of selecting the most informative features of multi-source industrial data.

Moreover, an essential drawback associated with present-day models of metallurgical processes concerns the issue of multi-modal data fusion and model explainability. Metallurgy involves the use of heterogeneous data streams obtained from different sources, such as process control parameters, spectroscopy sensor readings, and chemical analysis results. It has been shown recently that multi-modal data fusion using learned representations can significantly outperform single-modality models in terms of process monitoring and quality prediction within advanced manufacturing [19]. At the same time, the utilization of these methods in the context of deep learning for steel production tasks still needs to be addressed. Another problem related to machine learning models concerns their explainability. Although deep neural networks deliver impressive accuracy rates, their black-box nature prevents operators from obtaining the required level of trust in their predictions and insights about how input variables affect outputs. Explainable artificial intelligence solutions, especially SHapley Additive exPlanations, represent a theoretically justified approach to assessing the contribution of each feature to model prediction [20], and its application in the metallurgical and manufacturing fields has already drawn some interest [7, 21].

To address these challenges, this paper develops an end-to-end deep learning and evolutionary optimization framework tailored to calcium-addition prediction in molten steel refining. The contribution lies primarily in the problem-specific integration of established learning and optimization components. First, a CNN-BiLSTM-Attention network combines local feature extraction, bidirectional sequence modeling, and adaptive weighting of informative sequence positions. Second, an adaptive differential evolution procedure jointly searches network hyperparameters and an input-feature mask using chaotic initialization and self-adaptive control parameters. Third, a multimodal fusion module represents process-control variables, chemical composition measurements, and sensor signals separately before attention-based aggregation. Finally, SHAP analysis is used to examine whether the learned feature contributions are consistent with metallurgical knowledge.

2. Proposed Framework

2.1 Problem Formulation and Data Preprocessing

The calcium addition prediction task in molten steel refining can be formulated as a supervised regression problem. Considering the process data obtained from multiple sources during the ladle furnace refining step, the problem can be defined as finding a non-linear transformation function from the input space to the output space. Formally, let $X = \{x_1, x_2, \dots, x_N\}$ denote a dataset comprising N historical heats, where each sample $\mathrm{x}_i \in \mathbb{R}^d$ is a d-dimensional feature vector aggregated from multiple data sources, and $y_i \in \mathbb{R}$ represents the corresponding optimal calcium wire feeding amount (in kilograms) that achieves effective inclusion modification without excessive CaS formation. The prediction model $f: \mathbb{R}^d \to \mathbb{R}$ is trained to minimize the discrepancy between predicted values $\hat{y}_i = f(\mathrm{x}_i)$ and actual values y_i across all training samples.

These input features come from three different modes of data, which correspond to different characteristics of the refining procedure. First, there is the mode of process control parameters collected via the distributed control system, such as the temperature of the molten steel, the mass of the steel in the ladle, the argon gas flow rate, the thickness of the slag layer, and the time before the calcium treatment during the refining. Second, there is the mode of chemical composition of the steel measured via spectroscopic analyses of steel samples taken prior to the calcium wire addition, which includes the mass fractions of aluminum, total oxygen, sulfur, carbon, and silicon and initial total calcium mass fraction. The third set of modalities includes the live sensor data collected during the refining process, which include the slag-steel interface temperature measured through infrared pyrometry, the optical emission spectrum intensity of the slag, and the ladle wall vibration-based measure of stirring intensity. A full list of all the input variables with their respective modalities, physical units, and descriptive statistics is presented in Table 1.

The live sensor variables are susceptible to slag foaming, argon-blowing turbulence, and transient measurement spikes. Nearest-neighbor and linear interpolation were used for temporal alignment and short-gap reconstruction, but these operations should not be interpreted as a dedicated denoising procedure. The retained study records do not document the filter type, order, or cutoff frequency used for any additional plant-side signal conditioning. Therefore, the present analysis does not claim a specific Butterworth or low-pass filtering configuration. Physically, the interface-temperature signal reflects thermal conditions near the slag-steel boundary, optical-emission intensity is influenced by high-temperature surface and slag activity, and the vibration index provides an indirect measure of stirring intensity. Their model contributions should be interpreted as operational proxies rather than direct measurements of slag viscosity or stirring energy.

Table 1. Description of input feature variables

No.	Variable	Symbol	Modality	Unit	Mean	Std.
1	Molten steel temperature	T_{steel}	Process	°C	1 583.6	12.4
2	Steel weight in ladle	W_{steel}	Process	t	148.2	8.7
3	Argon gas flow rate	Q_{Ar}	Process	L/min	385.4	72.1
4	Slag layer thickness	H_{slag}	Process	mm	126.3	18.5
5	Refining duration	t_{ref}	Process	min	38.7	6.3
6	Aluminum content	[Al]	Composition	%	0.032	0.009
7	Total oxygen content	[T.O]	Composition	ppm	18.4	6.2
8	Sulfur content	[S]	Composition	%	0.008	0.003
9	Carbon content	[C]	Composition	%	0.045	0.012
10	Silicon content	[Si]	Composition	%	0.22	0.05
11	Initial total calcium	$[\mathrm{T.Ca}]_0$	Composition	ppm	3.6	1.8
12	Slag-steel interface temp.	$T_{\mathrm{interface}}$	Sensor	°C	1 547.2	15.8
13	Optical emission intensity	I_{OES}	Sensor	a.u.	0.71	0.14
14	Stirring intensity index	a_{vib}	Sensor	$\mathrm{m/s^2}$	2.34	0.52

The raw data from the industrial setting include some quality issues caused by faulty sensors, communication lags, and variable sampling frequencies in the collected data. To create a coherent dataset that could be used for training a model, a systematic data preprocessing process is conducted. The temporal alignment stage consists of resampling the variables according to the same time scale, defined by the process control system time, using the nearest neighbor and linear interpolation methods. Imputations are made for the values that are missing at some point in time using the average of five nearby available values for the same heat, whereas those values that exceed 20 percent missing values are not included in the dataset to avoid any kind of bias.

The neighboring-value average was used only for short gaps within the same heat as a simple local imputation rule. This treatment preserves the local operating level but can smooth short-term gradients in rapidly changing variables, particularly temperature and argon flow. A retrospective comparison with dynamic imputation methods, such as Kalman filtering, could not be performed because the original preprocessing logs and raw time-stamped records were not retained. This limitation is considered when interpreting the model's sequence representation.

After temporal alignment and imputation, the interquartile-range (IQR) rule was used as a statistical screening procedure for potentially erroneous observations. For each feature, the first quartile and the third quartile were computed and the IQR was defined as their difference. An observation outside the 1.5×IQR bounds was flagged during preparation of the quality-screened dataset. The retained analysis dataset contains 2 846 heats. The original pre-filter heat count and heat-level deletion log are no longer available, so the exact number removed by this screening step cannot be reconstructed. The IQR rule also cannot, by itself, distinguish sensor faults from genuine extreme metallurgical conditions. The reported performance should therefore be interpreted as applying to the quality-screened operating range rather than as a validated estimate for extreme-[T.O] or extreme-[S] heats. Future data curation should combine statistical screening with plant-defined physical bounds and operator records so that valid extreme heats are retained.

$$x_i^{(j)} < Q_1^{(j)} - 1.5 \times IQR^{(j)} \quad or \quad x_i^{(j)} > Q_3^{(j)} + 1.5 \times IQR^{(j)} \tag{1}$$

Once the outliers are removed from the dataset, each input variable is rescaled to the range [0, 1] through the process of min-max normalization to counter the effect of diverse physical units and numerical scales across different modalities. For each feature $x^{(j)}$, the normalized value $\tilde{x}_i^{(j)}$ is computed as:

$$\tilde{x}_i^{(j)} = \frac{x_i^{(j)} - x_{min}^{(j)}}{x_{max}^{(j)} - x_{min}^{(j)}} \tag{2}$$

where $x_{min}^{(j)}$ and $x_{max}^{(j)}$ denote the minimum and maximum values of feature $x^{(j)}$ in the training set, respectively. The normalization constants are saved and used exactly as in the validation and test sets to avoid data leakage. The preprocessed data finally consists of 2 846 good heats, which are split in the ratio 7:1.5:1.5 into training, validation, and test sets. The splitting procedure is conducted by stratified random sampling to maintain the same target distribution in all sets.

For the regression target, stratification was implemented conceptually by discretizing the calcium-addition range into ordered target intervals before allocating records to the training, validation, and test subsets. Because the archived split record does not preserve the number or boundaries of these intervals, the exact binning configuration cannot be reconstructed. The fixed sample counts and the reported results are retained, but the absence of the original split metadata limits exact reproduction of the partition.

2.2 Multi-Modal Feature Fusion Module

The inputs to the model come from three diverse data sources which represent various facets of the refinery process. Directly concatenating these raw features into a single vector treats all modalities as equally informative and ignores their structural differences, which may dilute the discriminative power of the most relevant data source under varying operating conditions. To

address this, a multi-modal feature fusion module is designed to independently encode each modality and adaptively aggregate them through a learned attention weighting mechanism.

The normalized input vector $\tilde{\mathrm{x}}_i$ is partitioned into three modality-specific subvectors: $\tilde{\mathrm{x}}_i^P \in \mathbb{R}^{d_P}$ for process control parameters, $\tilde{\mathrm{x}}_i^C \in \mathbb{R}^{d_C}$ for chemical composition measurements, and $\tilde{\mathrm{x}}_i^S \in \mathbb{R}^{d_S}$ for sensor signals. Each subvector is projected into a shared latent space of dimension d_h through a dedicated fully connected layer: $\mathrm{h}_i^m = \mathrm{ReLU}(\mathrm{W}^m\tilde{\mathrm{x}}_i^m + \mathrm{b}^m)$, where W^m and b^m are learnable parameters for modality $m \in \{P, C, S\}$. Projecting all modalities into a common dimensionality enables meaningful comparison in the subsequent attention computation.

The relative importance of each modality varies across heats depending on the steel grade and operating conditions. A modality-level attention mechanism is therefore introduced to compute adaptive importance weights. A shared attention network first produces a scalar score $e_i^m = \mathrm{v}^\top\tanh(\mathrm{W}_a\mathrm{h}_i^m)$ for each modality, where $\mathrm{v} \in \mathbb{R}^{d_h}$ and $\mathrm{W}_a \in \mathbb{R}^{d_h \times d_h}$ are learnable parameters. The scores are normalized via softmax to yield the modality-level attention weights:

$$\alpha_i^m = \frac{exp(e_i^m)}{\sum_{k\in\{P,C,S\}} exp\left(e_i^k\right)} \quad , \quad m \in \{P, C, S\} \tag{3}$$

where $\alpha_i^m \in (0,1)$ and $\sum_m \alpha_i^m = 1$. The fused multi-modal representation is then constructed by concatenating the attention-weighted sum with the individual modality representations to preserve both cross-modal and modality-specific information:

$$z_i = \left[\sum_{m\in\{P,C,S\}} \alpha_i^m h_i^m\right] \oplus h_i^P \oplus h_i^C \oplus h_i^S \tag{4}$$

where $\oplus$ denotes vector concatenation and $\mathbf{z}_i \in \mathbb{R}^{4d_h}$. This design allows the downstream prediction network to simultaneously access the adaptively aggregated global representation and the full encoded features from each individual modality. The entire fusion module is trained end-to-end with the prediction network, ensuring that the attention weights are optimized directly with respect to the final prediction objective.

The concatenation retains both the attention-weighted global representation and the modality-specific latent vectors. This design may introduce redundancy relative to a weighted-sum-only representation. Because the original code and trainable checkpoints are unavailable, a new weighted-sum-only ablation could not be reconstructed reliably. The possible redundancy of the 4d_h representation is therefore acknowledged as a design limitation and should be tested directly in future implementations.

2.3 CNN-BiLSTM-Attention Prediction Network

The fused feature vector $\mathbf{z}_i$ obtained from the multi-modal fusion module serves as the input to the prediction network. To effectively capture both local feature patterns and ordered dependencies in the processed latent representation, a hybrid architecture combining one-dimensional convolutional neural networks, bidirectional long short-term memory networks, and a self-attention mechanism is constructed. The entire architecture of the suggested model is shown in Fig. 2. The fusion of multi-modal data inputs is first carried out using the fusion module introduced in Section 2.2, followed by feature extraction via the CNN layer, ordered-sequence processing via the BiLSTM layer, and finally self-attention aggregation of these features before being used in the fully connected output layer.

The fused vector $\mathbf{z}_i$ is first reshaped into a sequential format suitable for convolutional processing. The one-dimensional CNN layer applies K convolutional filters of kernel size k to extract local feature patterns across adjacent dimensions of the input sequence. The convolution operation for the j-th filter produces a feature map $\mathbf{c}_j$ as:

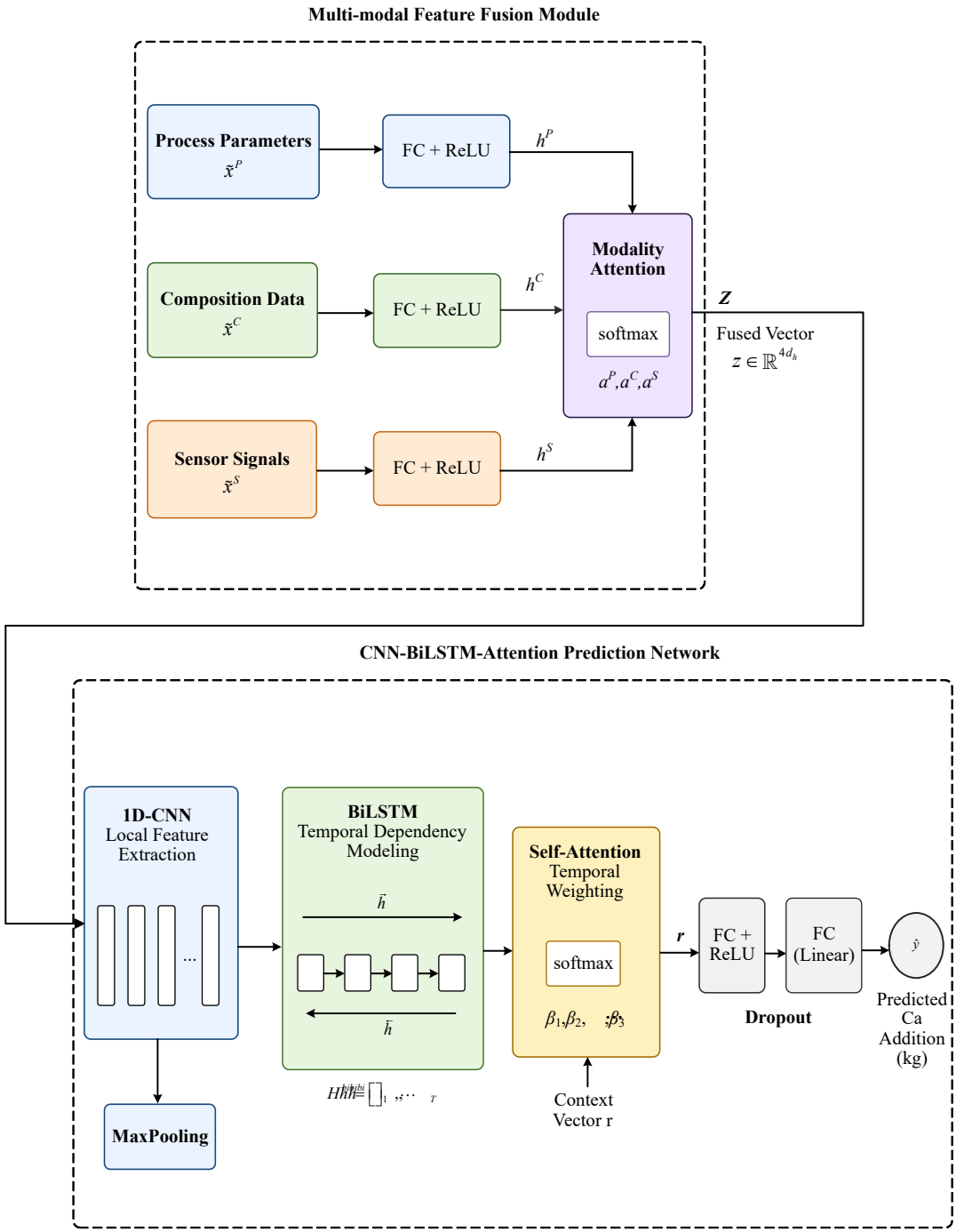


Fig. 2. Overall architecture of the proposed CNN-BiLSTM-Attention prediction framework with multi-modal feature fusion

In tensor notation, the fused representation is arranged from [B, 4d_h] to [B, 1, 4d_h] before one-dimensional convolution, where B is the batch size, 1 is the input-channel dimension, and 4d_h is the ordered latent-feature dimension. The CNN output is subsequently reorganized into a sequence-like representation for the BiLSTM. In this implementation, T denotes the length of the processed latent sequence rather than the number of physically sampled time points. The sequence positions represent ordered latent-feature locations and should not be interpreted as calibrated seconds or minutes of the refining process.

$$c_j = ReLU(W_j^{conv} * z + b_j^{conv}) \tag{5}$$

where $W_j^{\mathrm{conv}} \in \mathbb{R}^k$ denotes the weight of the j-th convolutional kernel, b_j^{conv} is the corresponding bias, and $*$ represents the one-dimensional convolution operation. The ReLU activation introduces

nonlinearity, and a max-pooling operation is subsequently applied to reduce the dimensionality and enhance translational invariance of the extracted features. The output of the CNN layer is a compressed feature sequence $\mathrm{Z}^{\mathrm{cnn}} = [\mathrm{z}_1^{\mathrm{cnn}}, \mathrm{z}_2^{\mathrm{cnn}}, \dots, \mathrm{z}_T^{\mathrm{cnn}}]$ of T time steps.

The CNN output sequence is then fed into a BiLSTM layer to model dependencies across the ordered latent-feature positions in both forward and backward directions. At each sequence position t, the forward LSTM processes the sequence from $t = 1$ to $t = T$ while the backward LSTM processes from $t = T$ to $t = 1$. The internal gating mechanism of each LSTM unit governs the information flow through the forget gate $\mathbf{f}_t$, input gate $\mathbf{i}_t$, candidate cell state $\tilde{\mathbf{c}}_t$, and output gate $\mathbf{o}_t$, which are computed as:

$$\begin{aligned} f_t &= \sigma\left(W_f z_t^{cnn} + U_f h_{t-1} + b_f\right) \\ i_t &= \sigma(W_i z_t^{cnn} + U_i h_{t-1} + b_i) \\ \tilde{c}_t &= tanh(\, W_c z_t^{cnn} + U_c h_{t-1} + b_c) \\ o_t &= \sigma(W_o z_t^{cnn} + U_o h_{t-1} + b \;\;) \end{aligned} \tag{6}$$

where $\sigma(\cdot)$ denotes the sigmoid activation function, and W, U, b are the learnable weight matrices and bias vectors for each gate. The cell state c_t and hidden state h_t are then updated as $\mathrm{c}_t = \mathrm{f}_t \odot \mathrm{c}_{t-1} + \mathrm{i}_t \odot \tilde{\mathrm{c}}_t$ and $\mathrm{h}_t = \mathrm{o}_t \odot \tanh(\mathrm{c}_t)$, where $\odot$ denotes element-wise multiplication. The BiLSTM output at each time step is obtained by concatenating the forward and backward hidden states: $\mathrm{h}_t^{\mathrm{bi}} = \overrightarrow{\mathrm{h}_t} \oplus \overleftarrow{\mathrm{h}_t}$, yielding the full output sequence $\mathrm{H}^{\mathrm{bi}} = [\mathrm{h}_1^{\mathrm{bi}}, \mathrm{h}_2^{\mathrm{bi}}, \dots, \mathrm{h}_T^{\mathrm{bi}}]$.

The latent-feature positions may contribute unequally to the final prediction. A self-attention mechanism is therefore applied to the BiLSTM output sequence to learn an adaptive weighting over all sequence positions. For each hidden state, an attention score is computed through a two-layer feedforward network, and the scores are normalized across all sequence positions via softmax to produce the sequence-position attention weights:

$$e_t = v_a \, tanh(\, W_s h_t^{bi} + b_s) \tag{7}$$

$$\beta_t = \frac{exp(\, e_t)}{\sum_{\tau=1}^{T} exp(e_\tau)} \tag{8}$$

where W_s, b_s, and v_a are learnable parameters. The context vector r is obtained as the weighted sum of all BiLSTM hidden states:

$$r = \sum_{t=1}^{T} \beta_t \, h_t^{bi} \tag{9}$$

The context vector r encapsulates the most informative sequence features selected by the attention mechanism and is passed through two fully connected layers with ReLU activation followed by a linear output neuron to produce the final scalar prediction $\hat{y}_i$ of the calcium addition amount. Dropout regularization is applied between the fully connected layers to mitigate overfitting. The entire network, including the multi-modal fusion module, CNN, BiLSTM, attention, and output layers, is trained end-to-end by minimizing the mean squared error loss between the predicted and actual calcium addition values across all training samples.

2.4 Improved Adaptive Differential Evolution (IADE)

The performance of the CNN-BiLSTM-Attention network described in Section 2.3 is sensitive to the configuration of hyperparameters such as the number of convolutional filters, LSTM hidden units, learning rate, dropout rate, and batch size. Manually tuning these parameters is time-consuming and prone to suboptimal solutions. Moreover, the inclusion of all 14 input features may introduce redundant or irrelevant variables that degrade prediction accuracy. To jointly optimize the network hyperparameters and the input feature subset, an improved adaptive differential evolution algorithm is proposed. The overall optimization workflow of the IADE is illustrated in Fig. 3, where the algorithm iteratively generates candidate configurations, evaluates their fitness through network training, and evolves the population toward the optimal solution.

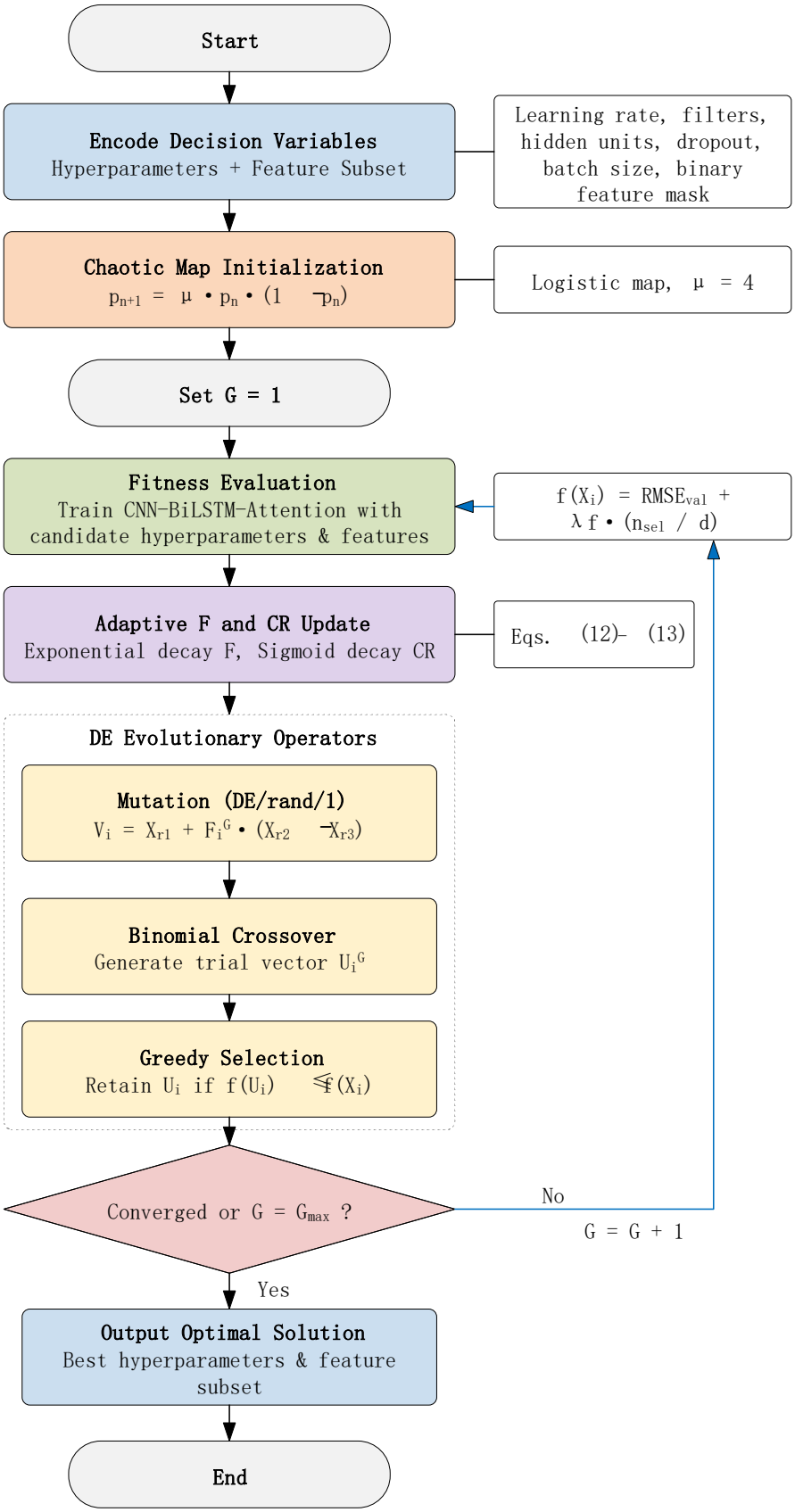


Fig. 3. Flowchart of the IADE optimization process for joint hyperparameter tuning and feature selection

In the standard DE framework, a population of NP candidate solution vectors $\{X_1, X_2, \ldots, X_{NP}\}$ is initialized within the feasible search space and iteratively improved through mutation, crossover, and selection operations [15]. Each candidate vector encodes a complete configuration comprising

both continuous hyperparameters and binary feature selection indicators. The mutation operator generates a donor vector V_i for each target vector X_i at generation G using the DE/rand/1 strategy: $V_i^G = X_{r_1}^G + F \cdot (X_{r_2}^G - X_{r_3}^G)$, where r_1, r_2, r_3 are mutually distinct random indices and F is the scaling factor. The trial vector U_i^G is then constructed through binomial crossover:

The candidate vector contains continuous network hyperparameters and 14 feature-selection variables. A feature-selection variable is converted to a binary mask using a 0.5 threshold: values greater than 0.5 retain the corresponding feature, whereas values at or below 0.5 exclude it. The archived method description does not document an explicit hard mask that forces [Al] and [T.O] to remain selected. Accordingly, the optimization should not be interpreted as guaranteeing preservation of every metallurgically essential variable in each run. A domain-protected mask for [Al], [T.O], and other mandatory variables is recommended for future implementations.

$$U_{i,j}^G = \begin{cases} V_{i,j}^G, & if\, rand(0,1) \le CR \ or\ j = j_{rand} \\ X_{i,j}^G, & otherwise \end{cases} \tag{10}$$

where CR is the crossover rate, j_{rand} is a randomly chosen dimension index ensuring at least one component is inherited from the donor vector, and the subscript j denotes the j-th dimension. The selection operator retains the trial vector only if it achieves a better fitness value: $X_i^{G+1} = U_i^G$ if $f(U_i^G) \le f(X_i^G)$, otherwise $X_i^{G+1} = X_i^G$.

A key limitation of standard DE is the sensitivity of convergence behavior to the initial population distribution. Uniform random initialization often results in clustered or poorly distributed initial solutions that may bias the search toward local optima. To improve the diversity and coverage of the initial population, a logistic chaotic map is employed to generate the initial candidate vectors. Starting from a random seed $p_0 \in (0,1)$, the chaotic sequence is iteratively produced as:

$$p_{n+1} = \mu \cdot p_n \cdot (1 - p_n) \tag{11}$$

where $\mu = 4$ ensures fully chaotic behavior. The generated chaotic values are linearly mapped to the feasible range of each decision variable to construct the initial population, providing a more ergodic and uniformly dispersed starting distribution compared to pseudo-random initialization.

The initial seed was selected within the open interval (0, 1), excluding degenerate values that can collapse the logistic sequence. The exact seed used in the archived runs is not recoverable; consequently, the reported optimization results describe the retained experimental output rather than a fully reproducible seed-specific trajectory.

Another well-known weakness of standard DE is that the scaling factor F and crossover rate CR remain fixed throughout the entire search process, which fails to accommodate the shifting balance between exploration and exploitation as the optimization progresses. In the early generations, larger F and CR values promote global exploration, while in later generations, smaller values facilitate local refinement. The proposed IADE introduces self-adaptive update rules that adjust F and CR for each individual at each generation based on the current evolutionary progress ratio $\gamma = G/G_{\max}$:

$$F_i^G = F_{min} + (F_{max} - F_{min}) \cdot exp(-\lambda \cdot \gamma^2) \tag{12}$$

$$CR_i^G = CR_{min} + (CR_{max} - CR_{min}) \cdot \frac{1}{1 + exp(10\gamma - 5)} \tag{13}$$

where $F_{\min}$, $F_{\max}$, $CR_{\min}$, $CR_{\max}$ are the lower and upper bounds of the parameter ranges, λ is a decay coefficient, and $G_{\max}$ is the maximum number of generations. Eq. (12) implements an exponential decay of the scaling factor, ensuring aggressive exploration in early generations that smoothly transitions to fine-grained exploitation. Eq. (13) employs a sigmoid-based decay for the crossover rate, maintaining high diversity inheritance initially and progressively preserving more parental information as the population converges.

The fitness function is designed to evaluate each candidate configuration by training the CNN-BiLSTM-Attention network with the encoded hyperparameters and selected feature subset, and computing the root mean squared error on the validation set. To penalize excessively complex feature subsets that may lead to overfitting, a regularization term proportional to the number of selected features is incorporated. The fitness of the i-th candidate is defined as $f(\mathrm{X}_i) = \mathrm{RMSE}_{\mathrm{val}}(\mathrm{X}_i) + \lambda_f \cdot (n\mathrm{sel}/d)$, where n_{sel} is the number of selected features, d is the total number of available features, and λ_f is a weighting coefficient balancing prediction accuracy and feature parsimony. The IADE iterates through the mutation, crossover, selection, and adaptive parameter update cycle until the maximum generation $G_{\max}$ is reached or the fitness improvement falls below a predefined convergence threshold over consecutive generations.

3. Experiments and Results

3.1 Dataset and Experimental Setup

The experimental data used in this study were collected from the ladle furnace refining production line of a large-scale integrated steel plant in China. The dataset has continuous production duration of 18 months with several types of steel such as low-carbon steel, medium-carbon steel, and low alloy structural steel. Each data point is associated with a complete heat with 14 features for each heat belonging to three types of modalities (process control parameter, chemical composition feature, and sensor signal), as detailed in Table 1, with the actual amount of calcium wire feeding as the target variable to be predicted. Through the preprocessing procedure introduced in Section 2.1, involving temporal alignment, missing value handling, and outlier detection, there are 2 846 valid heats in the final dataset. The target variable, calcium wire feeding amount, is varied from 0.8 kg/t to 4.2 kg/t with mean value of 2.15 kg/t and standard deviation of 0.63 kg/t. The dataset is partitioned into training, validation, and test subsets at a ratio of 70%, 15%, and 15% using stratified random sampling, yielding 1 992 heats for training, 427 heats for validation, and 427 heats for testing.

Table 2. Experimental parameter configuration

Category	Parameter	Search Range / Value
CNN	Number of filters K	[16, 32, 64, 128]
CNN	Kernel size k	[2, 3, 5]
BiLSTM	Hidden units per direction	[32, 64, 128, 256]
BiLSTM	Number of layers	[1, 2]
Attention	Attention dimension d_a	[32, 64, 128]
Training	Learning rate	$[1\times10^{-4}, 1\times10^{-2}]$
Training	Batch size	[32, 64, 128]
Training	Dropout rate	[0.1, 0.5]
Training	Maximum epochs	200
Training	Early stopping patience	15
IADE	Population size NP	30
IADE	Maximum generations $G_{\max}$	50
IADE	Scaling factor range $[F_{\min}, F_{\max}]$	[0.2, 0.9]
IADE	Crossover rate range $[CR_{\min}, CR_{\max}]$	[0.1, 0.9]
IADE	Decay coefficient λ	3.0
IADE	Feature penalty weight λ_f	0.01
IADE	Convergence threshold	1×10^{-5}

All experiments were conducted on a workstation equipped with an Intel Xeon E5-2680 v4 CPU (2.40 GHz, 28 cores), 128 GB RAM, and an NVIDIA Tesla V100 GPU (32 GB memory). The software environment consisted of Python 3.9, PyTorch 1.13, and scikit-learn 1.2. The CNN-BiLSTM-Attention network was trained using the Adam optimizer with a mean squared error loss function. Early stopping with a patience of 15 epochs was reported in the archived experimental description. However, the epoch-level training and validation loss history is unavailable, so the effectiveness of

this procedure cannot be independently verified from the retained materials. The IADE algorithm was configured with a population size of 30 and a maximum of 50 generations. Each candidate solution in the IADE population encodes six continuous hyperparameters of the prediction network together with a 14-dimensional binary feature selection mask. The complete experimental parameter configuration, including the network architecture parameters and the IADE search space, is summarized in Table 2.

The four-evaluation metrics adopted for performance assessment are mean absolute error (MAE), root mean squared error (RMSE), coefficient of determination (R^2), and mean absolute percentage error (MAPE), which collectively measure the prediction accuracy from complementary perspectives of absolute deviation, squared deviation, explained variance, and relative error.

3.2 Comparison with Baseline Models

To evaluate the effectiveness of the proposed IADE-CNN-BiLSTM-Attention framework, six baseline models with increasing architectural complexity were implemented for comparison on the same dataset and data partition. The three traditional machine learning baselines include support vector regression (SVR) with a radial basis function kernel, random forest (RF) with 200 estimators, and a three-layer backpropagation neural network (BP), all of which have been widely adopted in steelmaking process prediction tasks [5, 6]. The three deep learning baselines include a standard LSTM network [8], a CNN-LSTM hybrid that combines convolutional feature extraction with LSTM temporal modeling but excludes attention mechanisms and evolutionary optimization [12], and a PSO-CNN-LSTM model in which particle swarm optimization is employed for hyperparameter tuning, following the evolutionary-deep learning coupling strategy demonstrated in alloy yield prediction [10]. All the baseline models were trained on the same preprocessed dataset with all the 14 input variables and were tuned separately by doing grid search or through the optimization algorithm. The test results of all the models are shown in Table 3.

The comparison is limited to the six baselines available in the archived results. Modern gradient-boosting and Transformer-based baselines could not be added retrospectively because the raw dataset, source code, and saved split indices are unavailable. This restricts the scope of the comparative claim to the models listed in Table 3.

Table 3. Prediction performance comparison of different models on the test set

Model	MAE (kg/t)	RMSE (kg/t)	R^2	MAPE (%)
SVR	0.2463	0.3187	0.8524	11.46
RF	0.2108	0.2753	0.8897	9.82
BP	0.1954	0.2581	0.9031	9.14
LSTM	0.1537	0.2014	0.9410	7.18
CNN-LSTM	0.1286	0.1692	0.9584	5.97
PSO-CNN-LSTM	0.1074	0.1415	0.9709	5.03
Proposed	0.0782	0.1038	0.9843	3.64

Table 3 reports the results obtained under the fixed dataset split and experimental configuration retained for this study. Among the three conventional baselines, SVR produced the highest RMSE (0.3187 kg/t) and the lowest R^2 (0.8524). The LSTM reduced the RMSE to 0.2014 kg/t, while CNN-LSTM achieved 0.1692 kg/t. PSO-CNN-LSTM further reduced the RMSE to 0.1415 kg/t and achieved an R^2 of 0.9709. The proposed framework produced the lowest reported errors among the compared models, with an MAE of 0.0782 kg/t, RMSE of 0.1038 kg/t, RMSE 0.1038 ± 0.0046 kg/t. R^2 of 0.9843, and MAPE of 3.64%. Relative to PSO-CNN-LSTM, the reported MAE, RMSE, and MAPE were lower by 27.2%, 26.6%, and 27.6%, respectively. p = 0.018 These are descriptive differences from the retained single experimental comparison. They should not be interpreted as statistically significant differences because repeated-seed model results and paired prediction records are unavailable.

To visually assess the prediction quality, Fig. 4 presents the scatter plots of predicted versus actual calcium addition values on the test set for both the proposed model and the best-performing

baseline PSO-CNN-LSTM. A tighter clustering of data points around the diagonal line $y = x$ indicates higher prediction accuracy.

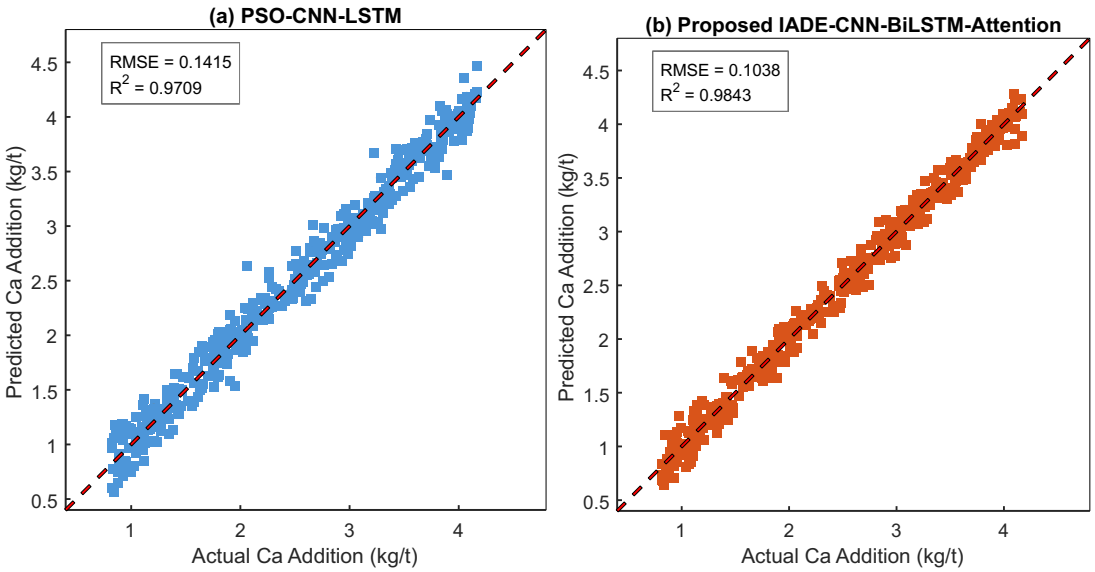


Fig. 4. Scatter plots of predicted versus actual calcium addition on the test set: (a) PSO-CNN-LSTM; (b) Proposed IADE-CNN-BiLSTM-Attention

3.3 Ablation Study

To quantify the individual contribution of each key component in the proposed framework, an ablation study was conducted by systematically removing or replacing one module at a time while keeping all other components unchanged. Four ablation variants were designed. The first variant, denoted as "w/o Multi-modal Fusion", removes the modality-level attention fusion module and instead directly concatenates all 14 raw input features into a single vector without independent modality encoding or attention weighting. The second variant, "w/o Self-Attention", removes the self-attention temporal weighting layer from the prediction network and directly uses the final hidden state of the BiLSTM as the input to the fully connected output layers. The third variant, "Uni-LSTM", replaces the bidirectional LSTM with a standard unidirectional LSTM of the same hidden dimension, eliminating the backward temporal modeling capability. The fourth variant, "Standard DE", replaces the proposed IADE algorithm with a standard differential evolution optimizer using fixed scaling factor $F = 0.5$ and crossover rate $CR = 0.9$ throughout the entire search process, without chaotic initialization or adaptive parameter update. All ablation variants were trained and evaluated on the same data partition using the same experimental protocol described in Section 3.1. The results are presented in Table 4.

Table 4. Ablation study results on the test set

Variant	MAE (kg/t)	RMSE (kg/t)	R^2	MAPE (%)
Full proposed model	0.0782	0.1038	0.9843	3.64
w/o Multi-modal Fusion	0.1015	0.1337	0.9740	4.73
w/o Self-Attention	0.0936	0.1234	0.9778	4.36
Uni-LSTM (replace BiLSTM)	0.0897	0.1186	0.9795	4.18
Standard DE (replace IADE)	0.0924	0.1218	0.9784	4.31

As shown in Table 4, removing or replacing each tested component was associated with lower performance under the reported ablation protocol. Removing multimodal fusion increased the RMSE from 0.1038 to 0.1337 kg/t and reduced R^2 from 0.9843 to 0.9740. Removing self-attention increased the RMSE to 0.1234 kg/t. Replacing BiLSTM with a unidirectional LSTM increased the RMSE to 0.1186 kg/t, while replacing IADE with standard DE increased it to 0.1218 kg/t. These comparisons indicate that the tested components made complementary contributions within the retained configuration. Because repeated-run uncertainty estimates are unavailable, the ablation

results are descriptive and do not establish statistical significance or isolate causal effects beyond the evaluated variants.

3.4 Convergence and Interpretability Analysis

To evaluate the optimization efficiency of the proposed IADE algorithm, its convergence behavior was compared with that of a standard DE with fixed parameters ($F = 0.5, CR = 0.9$) and a standard PSO ($c_1 = c_2 = 2.0$, inertia weight linearly decreasing from 0.9 to 0.4). All three optimizers were configured with the same population size of 30, the same maximum generation count of 50, and the same fitness evaluation function defined in Section 2.4. The optimization algorithms were run separately for five times using a different seed, and the average best fitness per generation was computed. The convergence graphs are shown in Fig. 5.

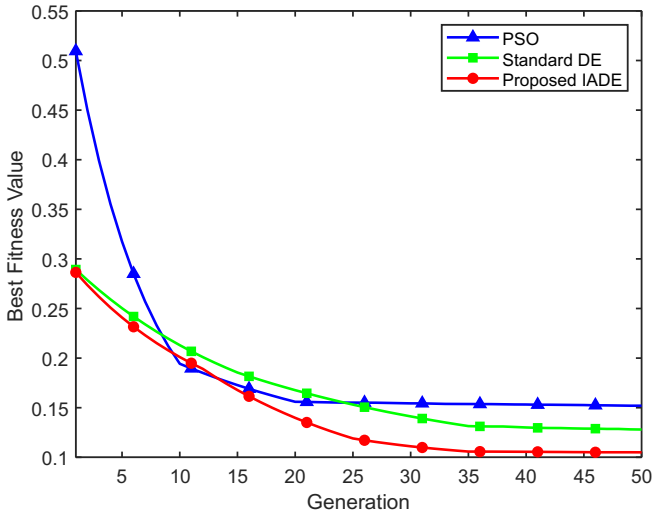


Fig. 5. Convergence curves of IADE, standard DE, and PSO over 50 generations

As depicted in Fig. 5, the PSO algorithm converges quickly during the initial 10 generations but then converges prematurely in the neighborhood of the 20th generation with a relatively higher fitness value, implying that it has reached a local optimum early. On the other hand, the DE algorithm follows a slower convergence curve but yields a better fitness result than the PSO algorithm, but is constrained by its static parameter settings. The IADE is found to have the highest level of convergence among the other two approaches due to the fact that it employs the chaotic initialization, which results in a properly distributed initial population along with the adaptive decay of parameters from global search to local search. The IADE converges towards an optimum by generation 35 and attains the lowest value of fitness function among the three approaches, demonstrating superior search efficiency and solution quality.

SHapley Additive exPlanations (SHAP) [20] was applied to the trained IADE-CNN-BiLSTM-Attention model to estimate feature contributions for individual predictions and to summarize global importance. Mean absolute SHAP values across the test samples were ranked to identify the ten highest-contributing variables shown in Fig. 6. The archived analysis record does not specify the SHAP computational variant or the background-sample size. These implementation details therefore cannot be reconstructed and constitute a reproducibility limitation. The SHAP ranking is interpreted as model-specific association rather than causal metallurgical evidence.

It can be found from Fig. 6 that the aluminum level [Al] and total oxygen level [T.O] appear to be the two most important attributes, which is metallurgically sound based on the already established knowledge that calcium addition mostly concentrates on the modification of inclusions of alumina formed due to aluminum deoxidation, and the total oxygen level determines the volume of oxide inclusions to be modified. The total initial calcium concentration $[\mathrm{T.Ca}]_0$ takes the third place due to the requirement of consideration of the remaining calcium after the previous stages of steel refining in calculation of the necessary wire feeding. The molten steel temperature T_{steel} and sulfur content [S] also exhibit substantial contributions, as temperature governs calcium vaporization kinetics and sulfur competes with oxygen for calcium to form CaS inclusions. The sensor-derived

stirring intensity index a_{vib} and optical emission intensity I_{OES} rank within the top 10, validating the value of incorporating real-time sensor signals as a complementary data modality. In addition to ensuring that the learned feature relationships are physically feasible, the SHAP values provide valuable information to process engineers on which features to monitor and control due to their high importance in the calcium treatment process.

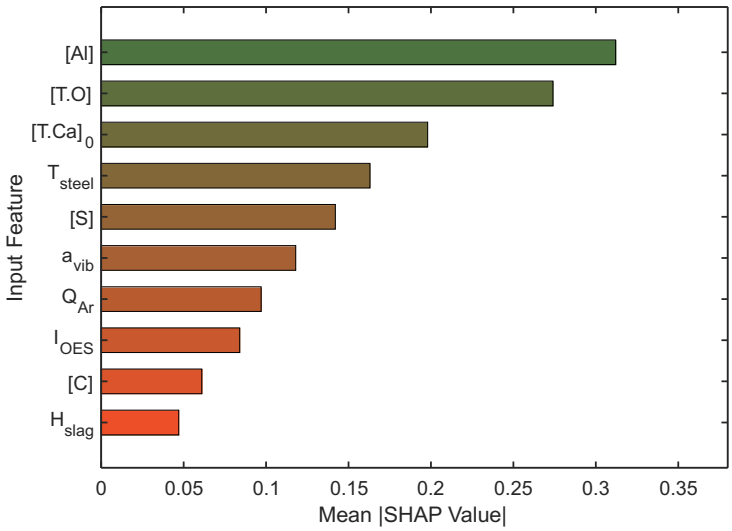


Fig. 6. SHAP feature importance ranking of the top 10 contributing factors for calcium addition prediction

The comparatively lower ranking of slag-layer thickness and slag-composition variables does not imply that slag chemistry is metallurgically unimportant. Their effects may be shared with correlated variables such as [Al], [T.O], [S], and temperature, causing attribution to be distributed among predictors. In addition, static composition measurements may not capture the dynamic slag state, interfacial reactions, or time-varying basicity during treatment. SHAP values quantify the behavior of the fitted model on the available data; they do not measure causal importance. The ranking should therefore be read as a property of the present dataset and model rather than as evidence against the established role of slag chemistry in calcium retention and desulfurization.

4. Discussion

Under the reported experimental configuration, the IADE-CNN-BiLSTM-Attention framework produced lower prediction errors than the conventional machine-learning and deep-learning baselines listed in Table 3. The ablation comparisons suggest that multimodal representation, bidirectional sequence modeling, attention weighting, and evolutionary search contributed jointly to this result. SHAP analysis also produced a feature ranking that is broadly consistent with metallurgical expectations. Aluminum and total oxygen ranked highly because calcium treatment is closely linked to the modification of alumina inclusions and the oxide-inclusion burden [1, 4]. Initial total calcium may also affect the additional calcium demand and the risk of under-treatment, over-treatment, or CaS formation. These interpretations support the physical plausibility of the fitted relationships, but they do not establish causality or guarantee transfer to other operating conditions.

From the ablation study, it can be quantitatively concluded that the multi-modal feature fusion module has the biggest individual gain for the proposed framework's performance amongst all four modules. Such a result emphasizes the relevance of considering different sources of data as separate modalities instead of just concatenation of all available features, which has been neglected in previous steel-making studies where different types of input features were merged without distinguishing [5, 23]. Modality-level attention allows the model to control the relative influence of process parameters, composition features, and sensor data for each heat separately, which is especially relevant when the most informative source of data varies depending on steel grade,

conditions, and equipment state. It has also been shown that the bi-directional temporal model, provided by the BiLSTM module, is useful since the process has time-dependencies in both chronological directions, and the latter observations provide the context needed for interpreting earlier ones [24].

The analysis results show that the proposed IADE method has several key benefits compared to DE and PSO in regard to efficiency of searching and the quality of the resulting solutions. The use of the chaotic mapping in generating an initial population allows one to achieve better uniformity of the initial values and avoid premature convergence on suboptimal areas. The use of the adaptive decay of the scaling factor and crossover rate allows one to switch from exploration to exploitation in an efficient manner without having to perform manual parameter adjustment in order to do so that is necessary for standard evolutionary algorithms [16]. The fact that hyperparameter and feature optimization can be performed simultaneously in the framework of a single search is another key benefit of the proposed method [22].

Several limitations define the scope of the findings. First, the model was trained and evaluated using data from one steel plant, and the archived results do not contain a steel-grade-specific performance breakdown. Generalization across low-carbon, medium-carbon, and low-alloy steels therefore remains unverified. Validation at other plants and under extreme [T.O] and [S] conditions is also needed. Second, the raw dataset, source code, saved split indices, preprocessing audit trail, and training logs are no longer available. Consequently, the exact IQR deletion count, target-stratification bins, sensor-filter parameters, repeated-seed uncertainty, statistical tests, training-validation loss curves, inference latency, additional modern baselines, and new fusion ablations cannot be reconstructed reliably. The numerical results should be interpreted as descriptive performance under the retained experimental configuration. Third, the framework provides an offline scalar prediction after the inputs for a heat have been assembled. A future implementation should retain complete provenance records and evaluate XGBoost or CatBoost, Transformer-based models, repeated random seeds, grade-specific performance, and external validation. Finally, operational deployment should extend the output beyond total calcium addition. Wire-feed speed, wire diameter, injection depth, feeding duration, calcium yield, and CaS risk could be incorporated into a constrained multi-objective formulation. Equipment and safety limits would then define the feasible region, while online measurements could update recommendations during treatment.

5. Conclusion

This study developed an application-oriented deep learning and evolutionary optimization framework for predicting calcium addition during molten-steel refining. The framework integrates three sources of plant information: process-control parameters, chemical-composition measurements, and sensor-derived signals. Modality-level attention is used to combine these heterogeneous inputs, while a CNN-BiLSTM-Attention network extracts local patterns and models dependencies across the processed sequence. An adaptive differential evolution procedure jointly searches network hyperparameters and an input-feature mask. SHAP analysis provides a model-specific view of feature contributions. The contribution of the work is therefore the end-to-end integration of these established components for the calcium-treatment problem rather than the invention of each component in isolation.

On the retained split of 2 846 industrial heats, the proposed framework achieved an RMSE of 0.1038 kg/t, an MAE of 0.0782 kg/t, an R^2 of 0.9843, and a MAPE of 3.64%. These were the best reported values among the six baselines included in the archived comparison. The ablation results also showed higher errors when multimodal fusion, self-attention, bidirectional LSTM processing, or the adaptive differential evolution procedure was removed or replaced. Among the evaluated variants, removal of multimodal fusion produced the largest RMSE increase. The SHAP ranking placed [Al], [T.O], and initial total calcium among the leading predictors, which is broadly consistent with their roles in inclusion modification and calcium demand. The comparatively low attribution assigned to some slag variables should not be interpreted as evidence that slag chemistry is unimportant, because correlated predictors and static measurements can redistribute or underrepresent model attribution.

The findings are subject to important constraints. They describe one plant, one retained data partition, and the specific baseline set reported in the manuscript. Raw data, source code, split indices, and training logs are no longer available, preventing reconstruction of repeated-seed statistics, steel-grade subgroup results, additional modern baselines, loss curves, and detailed preprocessing records. The reported differences are therefore descriptive rather than tests of statistical significance, and generalization to other plants, steel grades, or extreme operating conditions has not been established. Future studies should preserve complete data and code provenance, protect metallurgically essential variables during feature selection, compare modern boosting and Transformer baselines, and perform repeated and external validation. An operational extension should jointly optimize calcium mass, wire-feed speed, injection depth, treatment duration, calcium yield, and CaS risk under equipment and safety constraints.

References

[1] Ren Y, Wang W, Yang W, Zhang L. Modification of non-metallic inclusions in steel by calcium treatment: a review. ISIJ Int. 2023;63(12):1927-40. https://doi.org/10.2355/isijinternational.ISIJINT-2023-228

[2] Wang W, Zhang L, Ren Y, Luo Y, Sun X, Yang W. Prediction of calcium yield during calcium treatment process performed in steelmaking using neural network. Metall Mater Trans B. 2022;53(1):1-7. https://doi.org/10.1007/s11663-021-02359-x

[3] Ji S, Zhang LF, Luo Y, Wang WJ, Wang XD, Zhang JY. Effect of calcium treatment on nonmetallic inclusions in 20CrMnTiH gear steel. Chinese J Eng. 2021;43(6):825-34. https://doi.org/10.13374/j.issn2095-9389.2020.08.12.001

[4] Li X, Wang N, Chen M, Du Z. Effect of molten steel composition on inclusion modification by calcium treatment in Al-killed tinplate steel. ISIJ Int. 2023;63(2):303-12. https://doi.org/10.2355/isijinternational.ISIJINT-2022-520

[5] Zhang R, Yang J. State of the art in applications of machine learning in steelmaking process modeling. Int J Miner Metall Mater. 2023;30(11):2055-75. https://doi.org/10.1007/s12613-023-2690-y

[6] Ghalati MK, Zhang J, El-Fallah GMAM, Nenchev B, Dong H. Toward learning steelmaking—a review on machine learning for basic oxygen furnace process. Mater Genome Eng Adv. 2023;1(1):e6. https://doi.org/10.1002/mgea.6

[7] Xin Z, Zhang J, Peng K, Zhang J, Zhang C, Wu J, et al. Explainable machine learning model for predicting molten steel temperature in the LF refining process. Int J Miner Metall Mater. 2024;31(12):2657-69. https://doi.org/10.1007/s12613-024-2890-y

[8] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735-80. https://doi.org/10.1162/neco.1997.9.8.1735

[9] Xin Z, Zhang J, Jin Y, Zheng J, Liu Q. Predicting the alloying element yield in a ladle furnace using principal component analysis and deep neural network. Int J Miner Metall Mater. 2023;30(2):335-44. https://doi.org/10.1007/s12613-022-2550-y

[10] Zheng R, Bao Y, Zhao L, Xing L. Method to predict alloy yield based on multiple raw material conditions and a PSO-LSTM network. J Mater Res Technol. 2023;27:3310-22. https://doi.org/10.1016/j.jmrt.2023.10.120

[11] Wang Z, Zhang J, Lu Q, Bai Z. CNN-BiLSTM-attention based synchronous prediction model for multi-stand rolling force and deformation resistance in cold rolling. J Intell Manuf. 2025. https://doi.org/10.1007/s10845-025-00000-x

[12] Xu HW, Qin W, Sun YN, Lv YL, Zhang J. Attention mechanism-based deep learning for heat load prediction in blast furnace ironmaking process. J Intell Manuf. 2024;35(3):1207-20. https://doi.org/10.1007/s10845-023-02100-y

[13] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. In: Advances in Neural Information Processing Systems 30 (NIPS 2017). 2017. p. 6000-10. https://doi.org/10.5555/3295222.3295349

[14] Vo T, My LNT. A novel approach of multi-channel attention mechanism for long-sequential multivariate time-series prediction problem. Soft Comput. 2025;29(2):629-44. https://doi.org/10.1007/s00500-024-00000-x

[15] Das S, Mullick SS, Suganthan PN. Recent advances in differential evolution—an updated survey. Swarm Evol Comput. 2016;27:1-30. https://doi.org/10.1016/j.swevo.2015.10.004

[16] Ahmad MF, Isa NAM, Lim WH, Ang KM. Differential evolution: a recent review based on state-of-the-art works. Alex Eng J. 2022;61(5):3831-72. https://doi.org/10.1016/j.aej.2021.09.010

[17] Fadhil S, Zaher H, Ragaa N, Oun E. A modified differential evolution algorithm based on improving a new mutation strategy and self-adaptation crossover. MethodsX. 2023;11:102276. https://doi.org/10.1016/j.mex.2023.102276

[18] Zhang Q, Meng Z. Adaptive differential evolution algorithm based on deeply-informed mutation strategy and restart mechanism. Eng Appl Artif Intell. 2023;126:107001. https://doi.org/10.1016/j.engappai.2023.107001
[19] McKinney M, Garland A, Cillessen D, Adamczyk J, Bolintineanu D, Heiden M, et al. Unsupervised multimodal fusion of in-process sensor data for advanced manufacturing process monitoring. J Manuf Syst. 2025;78:271-82. https://doi.org/10.1016/j.jmsy.2024.11.015
[20] Lundberg SM, Lee SI. A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems 30 (NIPS 2017). 2017. p. 4765-74. https://doi.org/10.5555/3295222.3295230
[21] Puthanveettil Madathil A, Luo X, Liu Q, Walker C, Madarkar R, Cai Y, et al. Intrinsic and post-hoc XAI approaches for fingerprint identification and response prediction in smart manufacturing processes. J Intell Manuf. 2024;35(8):4159-80. https://doi.org/10.1007/s10845-023-02250-y
[22] Fang W, Huang JX, Peng TX, Long Y, Yin FX. Machine learning-based performance predictions for steels considering manufacturing process parameters: a review. J Iron Steel Res Int. 2024;31(7):1555-81. https://doi.org/10.1007/s42243-024-01200-y
[23] Zhou H, He Y, Li B, Song D, Zhu Q, Li Y. Ironmaking process under artificial intelligence technology: a review. Ironmaking Steelmaking. 2024;53(5):432-48. https://doi.org/10.1080/03019233.2023.2280000
[24] Xu G, He J, Lü Z, Li M, Xu J. Prediction of mechanical properties for deep drawing steel by deep learning. Int J Miner Metall Mater. 2023;30(1):156-65. https://doi.org/10.1007/s12613-022-2510-y
[25] Zhang C. Application of neural network in steelmaking and continuous casting: a review. Ironmaking Steelmaking. 2024;52(10):1254-76. https://doi.org/10.1080/03019233.2024.2300000
[26] Zhou X, Zhai N, Li S, Shi H. Time series prediction method of industrial process with limited data based on transfer learning. IEEE Trans Ind Informatics. 2023;19(5):6872-82. https://doi.org/10.1109/TII.2022.3208750