Supplementary Information

# Explainable machine learning model for predicting molten steel temperature in the LF refining process

**S1 Optimization Algorithms**

**(1) BO**

Snoek et al. [1] proposed the BO method. BO assumes that there is a functional relationship between the hyperparameters and the loss function to be optimized, and approximates the posterior distribution of the unknown objective function through a prior sample point [2]. The Gaussian process is used to train the data and learn the posterior distribution of the objective function in BO. The Gaussian process does not obtain the specific objective function value corresponding to each set of hyperparameters, but only gives the probability distribution of the objective function and obtains the expected mean and variance of the objective function in a certain region of the hyperparameter. Sampling at points with a large mean is called exploitation, and sampling at points with a large variance is called exploration. The acquisition function is utilized to balance exploitation and exploration, determining the location of the next sampling point. With the accumulation of sampling points, the posterior distribution of the objective function is constantly updated. On the basis of the new posterior distribution, new sampling points that maximize the acquisition function are continuously found and added to the sample set. The process is terminated when the maximum number of iterations is reached or the difference between the value of the objective function and the optimal solution is less than the threshold.

**(2)** **GWO**

Mirjalili et al.[3] proposed the GWO, a population intelligent optimization algorithm with simple structure, few parameters, easy implementation and strong global search ability. This algorithm simulates the social hierarchy and predatory behavior of grey wolf population in nature. Four types of grey wolves (α, β, δ, w) are used to simulate the social hierarchy, and the predatory behavior of wolves is simulated through the process of encircling prey, hunting, etc., as described below. **Figure S1** shows the flow chart of the GWO algorithm.



**Fig.S1 The flow chart of GWO algorithm**

1. Social hierarchy

The social hierarchy of grey wolves is constructed by computing the fitness of each wolf in the wolf pack. The wolves with the top three fitness values are designated as *α* Wolf, *β* Wolf, *δ* Wolf, while the rest are designated as *w* Wolf, as shown in **Figure S2(a)**. *α* Wolf, *β* Wolf and *δ* Wolf are responsible for leading the search, while *w* Wolf follows *α* Wolf, *β* Wolf and *δ* Wolf.

1. Encircling prey

In the hunting process, the grey wolf group will surround the prey first, and the mathematical expression of wolf position update during hunting behavior is shown in **Eq. (S1)**.

 (S1)

where, *D* represents the distance between the grey wolf and prey; *t* represents the number of iterations; *X*P(*t*) represents the position vector of the prey in the *t*-th iteration; *X*(*t*) represents the position vector of the grey wolf in the *t*-th iteration; *A* represents the convergence factor; *C* represents perturbation factors.

**Eq. (S2)** shows the calculation formula for *A* and *C*.

 (S2)

where, *a* is decreased linearly from 2 to 0 in the iterative process; *r*1 and *r*2 are random number in [0, 1].

1. Hunting

The hunting behavior is guided by *α* Wolf, *β* Wolf and *δ* Wolf, and the rest *w* Wolf changes its hunting position according to the positions of *α* Wolf, *β* Wolf and *δ* Wolf (*Xα*, *Xβ*, *Xδ*). The mathematical expression of hunting behavior is shown in **Eqs. (S3)-(S5)**. The position update of grey wolf in the hunting process is shown in **Figure S2(b)**.

 (S3)

 (S4)

 (S5)

where, the distance between *w* Wolf and *α* Wolf, *β* Wolf and *δ* Wolf is *Dα*, *Dβ* and *Dδ*, respectively. *X*(*t*+1) represents the updated position of the grey wolf. The grey wolf population approaches the prey by iterating continuously to update its position until the end of the iteration, and the hunting behavior is completed. Meanwhile, the optimal solution is obtained.



**Fig. S2 Diagram of leadership hierarchy and position update of gray wolves: (a)****Leadership hierarchy of grey wolves in nature; (b) The position update of grey wolf** **in the hunting process**

**S2** **ML Algorithms**

**(1) XGBoost**

XGBoost algorithm belongs to Boosting ensemble learning algorithm, which combines multiple weak learning models to get better results and make the combined model have stronger generalization ability [4]. Xgboost is not a simple combination of multiple weak learning models, but an addition strategy is used to iteratively add new weak learning models to fit the error of the last predicted result. When all the weak learning models are added, the ultimate prediction outcome is acquired by adding the prediction results of all the weak learning models [5]. To enhance the prediction accuracy of the XGBoost model while accelerating the convergence speed and effectively preventing overfitting, the second-order gradient of the loss function and the regular term is used in the XGBoost [6]. The final result of the XGBoost model can be expressed by **Eq. (S6)**.

 (S6)

where,  represents the predicted value of all *t* decision trees for sample *i*;  represents the predicted value of the former *t*-1 decision tree for sample *i*;  represents the predicted value of the *t* th decision tree for sample *i*.

1. **LGBM**

LGBM algorithm [7][8] is an effective decision tree gradient boosting method, which combines multiple weak regression tree models into a strong regression tree model (FT), as shown in **Eq. (S7)**. The Microsoft team made the following optimization based on XGBoost to speed up training and reduce memory consumption without losing computational accuracy. The Histogram algorithm and the leaf-wise tree growth strategy with max depth limitation were used in the LGBM algorithm, which not only improves training efficiency and space utilization, but also prevents overfitting. In addition, the gradient-based one-side sampling (GOSS) was utilized to reduce the data which have smaller gradients, and then improve the contribution for the calculated information gain. Meanwhile, the number of features was reduced by regrouping mutually exclusive features into bundles using exclusive feature bundling [8]. Considering this, this study also selected LGBM to establish the prediction model.

 (S7)

where *f*i is the *i*th weak regression tree.

**S3 SHAP**

Lundberg and Lee [9] proposed SHAP to provide explanations for ML models. The calculation of SHAP value is based on Shapley value that is a coalition game theory concept proposed by Shapley [10]. The impact of each feature on the predicted result is measured by calculating its contribution value, where the positive or negative value of the contribution value indicates the positive or negative influence. In addition, the importance of features increased as the absolute value of contribution value increased. Assuming the *i* th sample is *xi*; the *j* th feature of the *i* th sample is *xij*; the predicted value of the model for *xi* is *yi*; the base value of the model is *y*base; and the SHAP value of *xij* is *f*(*xij*). The relationship between the values can be expressed by **Eq. (S8)**.

 (S8)

**Eq. (S9)** shows the calculation formula of *f*(*xij*).

 (S9)

where, *N* is the set of all features, consisting of *M* feature; *S* is the subset taken from *N*, consisting of  feature;  represents the contribution resulting from the combined action of features included in subset S;  represents the contribution of feature *j* to the combined action;  represents the weight of .

**S4 Analysis of Feature dependency of different variables**

**Figure S3** shows the feature dependency graphs of different variables to further explore the influence rule of each variable on the prediction result. Figure S3 displays the variable value magnitude on the *X*-axis, the SHAP value of the variable value on the left *Y*-axis, and the magnitude of another interacting variable on the right *Y*-axis. The color scale ranges from blue to red, indicating the variable value changes from small to large. Fig.S3(a) shows the feature dependency graph of *X*1, and the SHAP value of *X*1 is mainly concentrated around 0, indicating that *X*1 has little influence on the prediction result. When *X*1 is greater than 70min, SHAP value of *X*1 is less than 0; with the increase of *X*1 value, SHAP value of *X*1 significantly decreases, that is, when the turnover time of ladle exceeds 70min, *X*1 has a negative effect on the prediction result; with the increase of turnover time of ladle, the negative effect of *X*1 increases. Cause analysis: The different thermal states of a ladle lead to different heat requirements for the lining's heat storage, resulting in different effects on the temperature drop of the molten steel. The thermal state of a ladle is directly affected by the turnover time of the ladle. With the turnover time of ladle increases, the heat exchange between the ladle furnace and the surroundings increases, leading to an increase in the temperature drop of molten steel. There is no significant interaction between *X*5 and *X*1. Fig.S3(b) shows the feature dependency graph of *X*2. When *X*2 is less than 152400kg, the SHAP value decreases with the increase of *X*2 value, and the predicted result decreases. When *X*2 is greater than 152400kg, the SHAP value increases with the increase of *X*2 value, and the predicted result increases. When *X*2 is between 150000kg-157000kg, the SHAP value is less than 0; with the increase of *X*6 value, the SHAP value of *X*2 decreases, that is, the temperature drop caused by argon bottom blowing increases with the increase of argon consumption, and the predicted result of molten steel temperature decreases. Cause analysis: Research indicates [11] that the temperature of the molten steel in the ladle becomes stratified after a period of time, and a low-temperature zone of molten steel forms at the bottom of the ladle. The temperature stratification of the molten steel can be alleviated by argon stirring to eliminate the formation of a low-temperature zone of molten steel at the bottom of the ladle. At this time, there is a large temperature difference between the temperature of the molten steel and the temperature of the ladle bottom, leading to increased heat loss and a larger temperature drop of molten steel.

Fig.S3(c) shows the feature dependency graph of *X*3, and *X*3 is positively correlated with its SHAP value. When *X*3 is less than 1575℃, the SHPA value is less than 0, that is, *X*3 has a negative effect on the prediction result; and with the increase of *X*3, the negative effect of *X*3 on the prediction result decreases. However, when *X*3 is greater than 1575℃, the SHAP value is greater than 0, that is, *X*3 has a positive effect on the prediction result; and with the increase of *X*3, the positive effect of *X*3 on the prediction result enhances. Meanwhile, the red dots are mainly distributed above the blue dots, suggesting that the larger the value of *X*5 is, the larger the SHAP value of *X*5 is, and the higher the prediction result will be at the same value of *X*3. Cause analysis: The LF refining process mainly relies on arc heating to increase the molten steel temperature. With an increase in the heating time, the amount of heat energy provided to the molten steel increases, leading to a further rise in the molten steel temperature. Fig.S3(d) shows the feature dependency graph of *X*4, and the SHAP value of *X*4 is mainly concentrated around 0, indicating that *X*4 has little influence on the prediction result. Overall, most SHAP values are less than 0, indicating that *X*4 has a negative effect on the prediction result. The SHAP value of *X*4 slightly decreases with the increase of *X*4, that is, the negative effect of *X*4 on the prediction result enhances. Cause analysis: The heat dissipation of the ladle shell and slag surface is related to the refining time in LF refining process. With the increase of LF refining time, the heat dissipation of the ladle shell and slag surface increases, resulting in a larger temperature drop of molten steel. In addition, the effect of *X*5 on the SHAP value and the molten steel temperature at the same *X*4 value is consistent with its effect at the same *X*3 value. Fig.S3(e) shows the feature dependency graph of *X*5, and *X*5 is positively correlated with its SHAP value. And the larger the value of *X*5 is, the larger the SHAP value of *X*5 is, and the higher the prediction result will be. There is no significant interaction between *X*5 and *X*2. Fig.S3(f) shows the feature dependency graph of *X*6. Overall, *X*6 is negatively correlated with its SHAP value. And when *X*6 is greater than 3×104NL, the negative effect of *X*6 on the prediction result enhances with the increase of *X*6. The effect of *X*5 on the SHAP value and the prediction result at the same *X*6 value is consistent with its effect at the same *X*3 value.



**Fig.S3** **Feature dependency graphs of different variables**

**References**

1. J. Snoek, H. Larochelle, and R.P. Adams, Practical Bayesian optimization of machine learning algorithms. [in] *Proceedings of the 26th Annual Conference on Neural Information Processing Systems*, Lake Tahoe, Nevada, 2012, p. 2951.
2. Y.Z. Li, Z.Y. Wang, Y.L. Zhou, and X.Z. Han, The improvement and application of Xgboost method based on the Bayesian optimization, *J. G. Univ. Technol.*, 35(2018), No. 1, p. 23.
3. S. Mirjalili, S.M. Mirjalili, and A. Lewis, Grey wolf optimizer, *Adv. Eng. Softw*., 69(2014), p. 46.
4. T.Q. Chen and C. Guestrin, Xgboost: A scalable tree boosting system. [in] *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, California, USA, 2016, p. 785.
5. W. Deng, Y.J. Mei, Y. Li, Y.X. Guo, T. Kang, and L. Ren, Daily peak load forecasting and heavy overload warning method of distribution transform-er based on BOXGBoost, *P. CSU-EPSA*, 2023, Online. DOI: 10.19635/j.cnki.csu-epsa.001295.
6. B.C. Zhao, J.J. Ma, L. Cui, W.P. Luan, and J. Zhu, Photovoltaic anomaly detection based on improved VMD-XGBoost-BILSTM hybrid model, *Comput. Eng.*, 2023, Online. DOI: 10.19678/j.issn.1000-3428.0067583.
7. B. Noam, K. Tal, W. Omer, Z. David, R. Daphna, L. Sigal, K. Noa, L.P. Maya,W. Adina, I.L.R. Caroline, M. Cristina, V. Alessia, F. Mario, D.S. Tim, I.D. The, A. Jerzy, W.F. Paul, P. Oluf, S. Eran, A reference map of potential determinants for the human serum metabolome, *Nature*, 588(2020), p. 135.
8. G.L. Ke, Q. Meng, T. Finley, T.F. Wang, W. Chen, W.D. Ma, Q.W. Ye, and T.Y. Liu, LightGBM: A highly efficient gradient boosting decision tree. [in] *Proceedings of the 31st Annual Conference on Neural Information Processing Systems*, Long Beach, California, USA, 2017, p. 3147.
9. S.M. Lundberg and S.I. Lee, A unified approach to interpreting model predictions. [in] *Proceedings of the 31st Conference and Workshop on Neural Information Processing Systems*, Long Beac, California, USA, 2017, p. 4765.
10. L. Shapley, *Contributions to the Theory of Games II*, eds. H. Kuhn and A. Tucker. Princeton University Press, 1953.
11. J. Li, *LF Refining Technology*, Metallurgical Industry Press, Beijing, 2012.