Integrating Stacking-Ensemble Feature Selection with GRU-Based Self-Supervised Learning for Precision and Uncertainty-Aware Steam Flow Forecasting

Hongxia Zhou, Fang Yuan, Li Guo, and Min Gan

***Abstract*—Steam flow is a crucial indicator for assessing the effectiveness of municipal solid waste incineration (MSWI), a major municipal waste treatment technique, and power generation performance. However, there remain some challenges in accurately predicting steam flow. Traditional single feature selection methods often fail to capture essential relationships by analyzing data from a single perspective, limiting model performance. Additionally, supervised learning often suffers from its reliance on task-specific feature engineering, which is overly sensitive to data distribution and demands significant domain expertise. To address these issues, we propose a predictive framework that combines a stacking ensemble of feature selection with self-supervised learning (SSL). The stacking ensemble strategy captures variable relationships from diverse perspectives, which enhances the robustness of feature selection. Meanwhile, SSL helps the model learn general feature representations and balance short- and ultra-short-term dependencies. Experimental results show that the proposed method outperforms existing approaches in both point prediction and prediction interval tasks. It improves generalization, uncertainty quantification, and robustness for complex industrial data, thereby supporting better risk assessment and decision-making.**

***Index Terms*—Steam flow prediction; Self-supervised learning; Ensemble feature selection; Prediction interval.**

I. Introduction

In recent years, with the rapid economic growth and accelerated urbanization in China, waste management has become an increasingly pressing issue. As shown in Fig. 1, by 2022, the annual volume of municipal solid waste (MSW) collected in Chinese cities exceeded 200 million tons, with a continuous upward trend, presenting significant challenges to urban environmental management and sustainable development. Compared to traditional landfilling, municipal solid waste incineration (MSWI) technology not only effectively reduces waste volume but also converts thermal energy into electrical energy through high-temperature combustion, enabling resource recovery [1]. MSWI has increasingly become the primary method for MSW treatment [2]. In this process, steam flow serves as the core of energy conversion, directly reflecting the combustion stability of the incinerator and the efficiency of power generation [3]. Therefore, accurately predicting steam flow is crucial for ensuring stable incinerator operation and enhancing power generation quality.

Machine learning (ML), as a data-driven approach, has become an essential tool for MSWI prediction tasks due to its strong data processing and pattern recognition capabilities [4]. Common methods include multiple linear regression [5], support vector regression (SVR) [6], [7], artificial neural networks (ANN) [8], and gradient boosting regression trees (GBRT) [9]. Xu et al. [10] proposed an industrial steam flow prediction model using grid search-optimized SVR to improve accuracy. However, these methods have notable limitations: linear models are ineffective at capturing nonlinear features, while ANN and SVR incur high computational costs when processing large, high-dimensional datasets and require extensive hyperparameter tuning.

Deep learning (DL) has become a widely applied method in industrial predictive modeling [11]. Models such as Long Short-Term Memory networks (LSTM) [12], Gated Recurrent Units (GRU) [13], Transformer [14], and its variants leverage deep neural networks to capture both short-term and long-term dependencies in data. These models effectively address the complex nonlinear relationships in industrial tasks and are thus frequently employed for MSWI prediction. Tian et al. [15] proposed a DL model combining one-dimensional convolutional neural networks (1D-CNN) and LSTM to predict differential thermal analysis (DTA) data during the fine screening incineration process, utilizing TensorBoard for model unit visualization. Hu et al. [16] devised a multi-step prediction approach integrating Principal Component Analysis (PCA) with Pyraformer [17] to forecast the main steam flow in MSWI systems. Transformer and its variants excel in long-sequence modeling but require substantial amounts of data and computational resources. In contrast, GRU models, with a simpler architecture, deliver comparable performance in many time-series tasks while achieving higher efficiency and lower resource demands [18]. Additionally, the recursive structure of GRU captures sequence dependencies without the need for explicit positional encoding, making it a practical choice for real-time predictions in resource-constrained industrial environments.

DL models are typically built within the supervised learning (SL) framework but often struggle with challenges

Manuscript received April 27, 2025; revised July 9, 2025.

Hongxia Zhou is a postgraduate student in the College of Computer Science and Technology, Qingdao University, Shandong Province, Qingdao, 266071, China. (e-mail: zhouhongxia@qdu.edu.cn).

Fang Yuan is a Ph.D. student in the College of Computer Science and Technology, Qingdao University, Shandong Province, Qingdao, 266071, China (e-mail: yuanfang2@qdu.edu.cn).

Li Guo is an associate professor in the College of Computer Science and Technology, Qingdao University, Shandong Province, Qingdao, 266071, China (e-mail: ally_guo@qdu.edu.cn).

Min Gan is a professor in the College of Computer Science and Technology, Qingdao University, Shandong Province, Qingdao, 266071, China (e-mail: aganmin@aliyun.com).

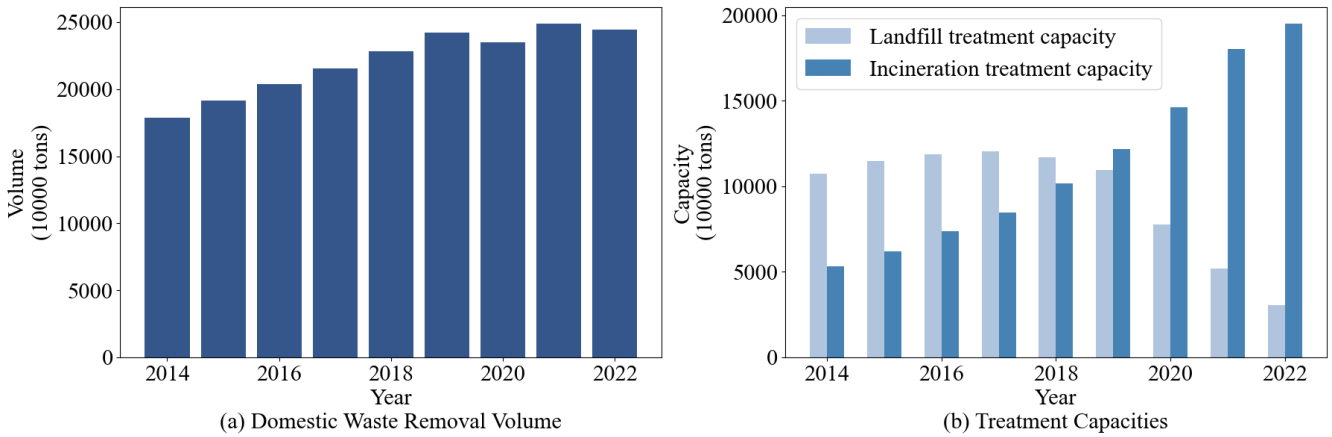

Fig. 1. China's waste removal and treatment capacity in 2022. Data from China's National Bureau of Statistics: (a) waste removal reached 244.447 million tons by 2022, and (b) waste treatment shifted towards incineration, with a capacity of 195.02 million tons, compared to 30.432 million tons for landfill.

such as diverse data distributions and noise in real-world conditions. These models also rely heavily on task-specific feature engineering, limiting their generalization ability. Recently, self-supervised learning (SSL) has emerged as an effective method for extracting high-level features from unlabeled data [19], though its application in MSWI prediction remains underexplored. SSL enables models to learn meaningful representations through pretraining tasks, such as predicting data segments or reconstructing masked information. These representations can be effectively utilized for downstream tasks, including prediction and classification, thereby enhancing the model's generalization capabilities [20]. For example, the BERT model [21] achieves excellent performance in downstream tasks by pretraining on unlabeled text data. In MSWI prediction, SSL can capture temporal dependencies and dynamic features through tasks like forecasting future steps or reconstructing missing data, effectively addressing noise and distribution shifts while enhancing model generalization and applicability [22].

Furthermore, the operating conditions of waste incineration are complex, involving dozens of related factors. Most current waste incineration forecasting methods rely on single feature selection approaches to capture global features. However, such methods typically fail to comprehensively evaluate the multidimensional associations between features and target variables, limiting the ability of subsequent predictive models to select richer and potentially valuable information. For example, Xu et al. [10] employed the Spearman correlation coefficient to identify overall monotonic relationships in the data. Similarly, Hu et al. [16] utilized Principal Component Analysis (PCA) for dimensionality reduction. However, the variance-maximization assumption of PCA restricts its ability to capture nonlinear correlations, thereby constraining the diversity of feature representations. Consequently, predictive models that rely on single feature selection approaches struggle to fully exploit the latent information within the data, hindering further improvements in model performance.

To address these challenges, this paper presents a Stacking Ensemble of Feature Selection and GRU-based SSL (SEFS-GSSL) method tailored for steam flow prediction. This method integrates multiple feature selection techniques to construct a more comprehensive feature set, leveraging stacking and weighted assignment strategies to enhance model robustness and predictive performance. SEFS-GSSL also employs target-masked self-supervised pretraining, enabling it to more accurately capture temporal dependencies and general feature representations. This approach significantly improves the accuracy and generalization of steam flow prediction while providing valuable insights for uncertainty quantification and risk management. The main contributions of this work are as follows:

- We integrate multiple feature selection methods by stacking and weighted assignment strategies to enhance model robustness and predictive performance in capturing complex feature patterns, reducing over-reliance on any single approach.
- We propose a GRU-based SSL strategy with a target variable masking mechanism. Through pre-training and fine-tuning, the model effectively learns dependencies and feature distributions from historical data while adapting to variations across different time periods, significantly improving the accuracy and generalizability of steam flow predictions in complex industrial systems.
- To quantify prediction uncertainty, we employ Gaussian-based error probability density estimation to construct prediction intervals at multiple confidence levels. This approach improves result interpretability and provides robust information for risk management and decision-making in industrial systems.

II. System Background

The core of the MSWI technology lies in efficiently converting the energy in waste into electrical power while ensuring environmentally safe treatment of exhaust gases and residues. As shown in Fig. 2, the MSWI process primarily includes key stages such as incineration, steam generation, power generation, and exhaust gas treatment. The following provides a detailed overview of these stages.

MSW is typically delivered to the incineration power plant via sealed transportation vehicles, where it is first weighed and counted to ensure accurate measurement. Subsequently,

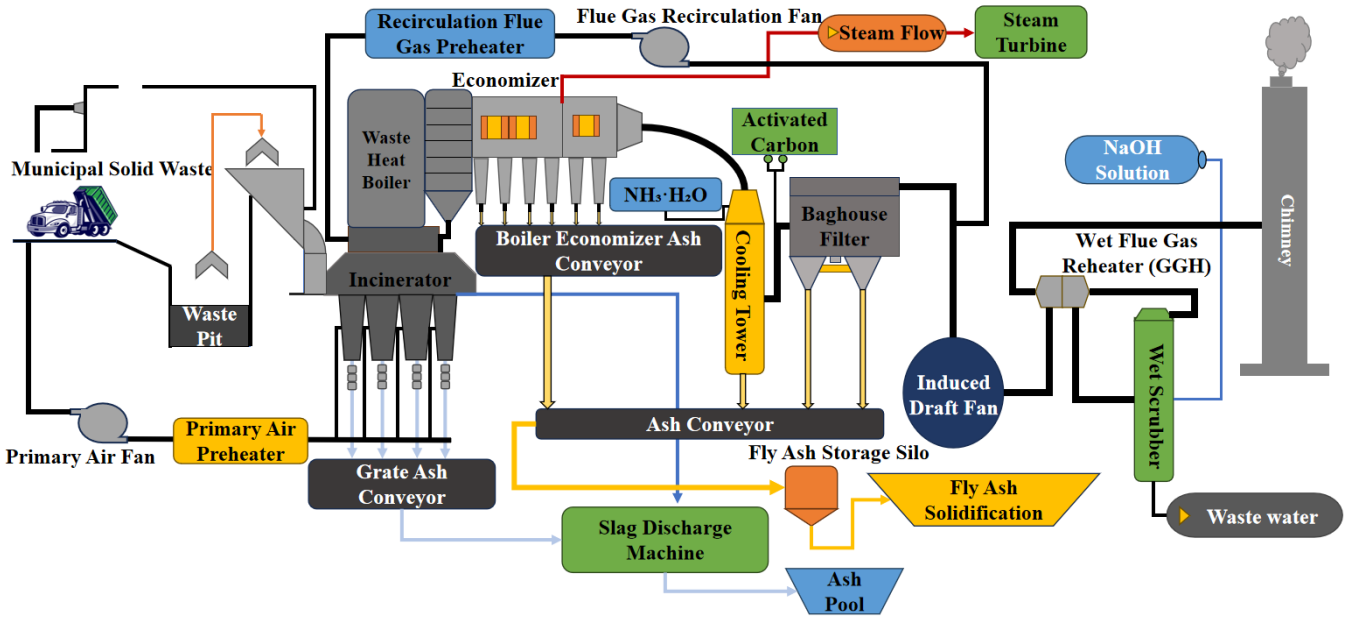

Fig. 2. Industrial process flowchart of MSWI technology.

the waste is transferred through enclosed conveyor systems into the plant's storage area, referred to as the waste pit, awaiting further processing. Mechanical grappling devices then load the waste into the incinerator for combustion treatment.

Inside the incinerator, waste undergoes high-temperature combustion. The incinerator's temperature is generally maintained between 850°C and 1000°C to ensure complete combustion and minimize residual materials. During combustion, significant thermal energy is released, accompanied by high-temperature flue gases, typically exceeding 850°C. These flue gases are directed from the incinerator to the waste heat boiler, a crucial component in the incineration-to-power process, designed to recover heat from the flue gases and convert it into high-temperature, high-pressure steam.

The waste heat boiler typically contains multiple heat exchange tubes filled with water. As the flue gases pass through these tubes, they transfer heat to the water within, rapidly increasing its temperature and causing vaporization at high pressure. This process essentially transforms the thermal energy of the flue gases into the internal energy of steam, creating a stable steam flow [23]. Then the generated steam is directed to a downstream turbine apparatus, where it drives the rotation of turbine blades, subsequently generating electrical energy. To further enhance efficiency, the boiler is often equipped with an economizer [24] to recover additional heat from the flue gases, preheating the boiler feedwater and improving the overall thermal efficiency of the system.

After completing the power generation process, the remaining flue gases move on to the next treatment phase. Initially, the gases pass through a bag filter to remove fine particulates and dust generated during combustion. The gases then enter a wet scrubber, where they are sprayed with an NaOH solution to neutralize acidic substances such as sulfur dioxide. Following purification, the exhaust gases are finally discharged through the chimney by induced draft fans.

The incineration process also generates slag and fly ash, which are collected and treated through specialized transport systems. Slag, mainly consisting of unburned mineral matter, can generally be recycled as building materials or roadbed materials. Fly ash, however, requires special treatment, such as solidification, to ensure safe disposal.

III. Methodology

The overall architecture of the proposed steam flow prediction network is illustrated in Fig. 3. This framework first applies a stacking ensemble feature selection method to identify key variables influencing steam flow from the raw input data. Then, the extracted multivariate data is fed into a GRU-based SSL model with a masking mechanism for pre-training. The pre-trained model is subsequently used for downstream steam flow prediction tasks, with prediction intervals generated based on point prediction errors. Details of the model's implementation are provided in subsequent sections.

A. Feature selection method based on SEFS

SEFS is an advanced ensemble learning method that integrates multiple feature selection techniques with weighted feature evaluation to not only select the most relevant features from the data but also preserve the unique advantages of each individual approach, such as minimizing feature redundancy. Specifically, we employ three distinct types of feature selection methods: (1) Filter method: Maximal Relevance Minimal Redundancy (mRMR), which selects the most relevant features while minimizing redundancy based on mutual information; (2) Model-based method: Random Forest (RF), which ranks features by their importance to the target variable; (3) Embedded method: Lasso Regression, which utilizes L1 regularization during model training to simultaneously compress and select features.

After obtaining feature subsets $S_{\text{mRMR}}, S_{\text{RF}}, S_{\text{Lasso}}$ generated by each feature selection method, we perform

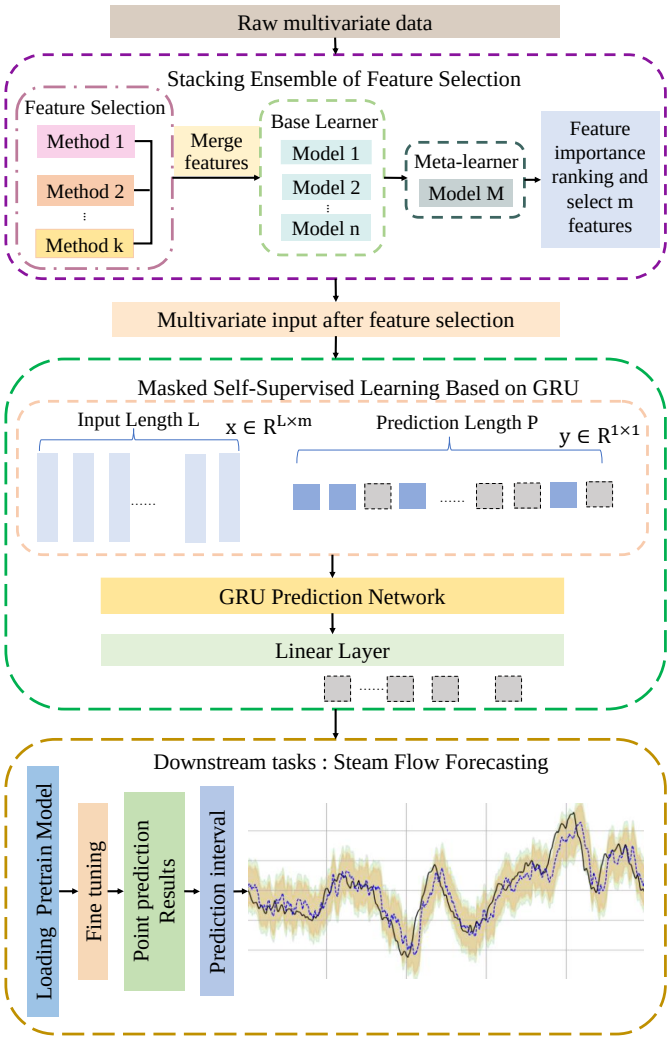

Fig. 3. The SEFS-GSSL framework for steam flow forecasting.

a merging process to eliminate redundant features. The resulting feature set is defined as follows:

$$S_{\text{combined}} = (S_{\text{mRMR}} \cup S_{\text{RF}} \cup S_{\text{Lasso}})_{\text{unique}} \tag{1}$$

where $(\cdot)_{\text{unique}}$ denotes the unique feature set obtained after eliminating duplicate features, ensuring that S_{combined} contains the distinct and meaningful features selected by each method, comprehensively capturing the diversity in the data to enhance robustness and adaptability in the model.

After obtaining the combined feature set S_{combined}, SEFS method utilizes a stacking ensemble strategy to train multiple base learners on this feature set. We selected three base learners: Ridge regression, Decision Tree regression, and XGBoost. Each base learner has unique advantages, enabling them to handle both linear and nonlinear relationships in the feature set and to capture data patterns from different perspectives. To reduce model complexity, we employ a simpler Linear regression model as the meta-learner. The loss function of the meta-learner can be defined as follows:

$$L = \sum_{i=1}^{N} \left(Q_i - \hat{Q}_i\right)^2 \tag{2}$$

$$\hat{Q}_i = w_{\text{Rd}}\hat{Q}_{\text{Rd},i} + w_{\text{DT}}\hat{Q}_{\text{DT},i} + w_{\text{XGB}}\hat{Q}_{\text{XGB},i} \tag{3}$$

where Q_i denotes the true steam flow value for the i-th sample, $\hat{Q}_{\text{Rd},i}$, $\hat{Q}_{\text{DT},i}$, and $\hat{Q}_{\text{XGB},i}$ represent the predicted values of the i-th sample obtained through the corresponding base learner. w_{Rd}, w_{DT}, and w_{XGB} represent the weights assigned to the output of each base learner, reflecting the contribution of each learner to the final prediction. These weights are optimized by minimizing the loss function L, enabling an optimal weighted combination of base learner predictions in the ensemble model, thereby enhancing the model's overall predictive performance and generalization ability.

Finally, SEFS integrates the importance scores of each base learner with their respective weights to obtain the final comprehensive feature importance score. The overall feature importance $I(X_j)$ is calculated by summing the weighted importance scores from each base learner, as shown below:

$$\begin{aligned} I(X_j) = {} & w_{\text{Rd}} \cdot I_{\text{Rd}}(X_j) \\ & + w_{\text{DT}} \cdot I_{\text{DT}}(X_j) \\ & + w_{\text{XGB}} \cdot I_{\text{XGB}}(X_j) \end{aligned} \tag{4}$$

where $I_{\text{Rd}}(X_j)$, $I_{\text{DT}}(X_j)$, and $I_{\text{XGB}}(X_j)$ represent the importance scores of feature X_j in Ridge regression, Decision Tree regression, and XGBoost, respectively.

Overall, SEFS constructs a diverse feature set through multiple feature selection methods and further integrates it with an ensemble of weighted base learners. This weighted ensemble strategy not only improves the accuracy of feature importance evaluation and predictive performance but also significantly enhances the model's robustness and generalization capabilities.

B. GRU enhanced by SSL

To address the challenges posed by data noise and distribution diversity in MSWI prediction, this paper proposes a steam flow prediction approach based on SSL, with GRU serving as the core prediction model. SSL enables the model to learn from unlabeled data by introducing a masking mechanism, where target variables are partially obscured. Then the model is trained to reconstruct these masked portions using surrounding context, thereby effectively capturing temporal dependencies and learning robust feature representation from the data itself.

The foundational GRU model, an improved variant of the recurrent neural network (RNN) [25], is designed to overcome the gradient vanishing problem common in traditional RNNs when processing long sequential data. GRU utilizes two gating units—the reset gate and update gate—to dynamically control information flow through selective updating and forgetting [26]. This structure allows GRU to capture both long-term and short-term dependencies in time series data. The hidden state update equations for GRU are as follows:

$$Z_t = \sigma(W_z h_{t-1} + U_z x_t) \tag{5}$$

$$R_t = \sigma(W_r h_{t-1} + U_r x_t) \tag{6}$$

$$\tilde{h}_t = \tanh(W_h(h_{t-1} \odot r_t) + U_h x_t) \tag{7}$$

$$h_t = Z_t \odot \tilde{h}_t + (1 - Z_t) \odot h_{t-1} \tag{8}$$

where x_t is the input to the GRU model at time step t, h_t is the hidden state at time t, σ is the sigmoid activation function, and $\odot$ represents element-wise multiplication. The matrices W_z, W_r, and W_h apply transformations on the previous hidden state h_{t-1}, while U_z, U_r, and U_h are weight matrices that apply transformations on the current input x_t. Here, Z_t and R_t are the update and reset gates, respectively.

Within the GSSL framework, the GRU model serves as the core predictive component, designed to reconstruct masked target values in time series data. In this context, let $Q = [Q_1, \ldots, Q_t, \ldots, Q_T]$ represent the true steam flow values over T time steps. The objective is to reconstruct the value Q_t at time step t using contextual information. The objective function, designed to minimize the error between the predicted and true values, is expressed as:

$$L_{\text{mask}} = \sum_{t \in M} loss(Q_t, \hat{Q}_t) \tag{9}$$

where M denotes the set of masked time steps, and $\hat{Q}_t$ is the model's prediction of Q_t. Here, $loss(\cdot)$ quantifies the error between the predicted value $\hat{Q}_t$ and the true value Q_t. By minimizing this loss, the GRU model learns temporal dependencies from unlabeled data.

This process enables the model to better learn the intrinsic patterns and temporal dependencies of the sequence, enhancing its ability to capture complex dependencies and increasing robustness to data noise. Consequently, it improves the accuracy of steam flow predictions in downstream tasks.

C. Gaussian distribution and prediction interval

The Gaussian distribution, also known as the normal distribution, is a fundamental distribution widely used in probability and statistics, particularly in constructing probability density models for modeling errors and uncertainties [27]. In the context of steam flow prediction, we assume the prediction error of all samples at time t, denoted as Err_t, follows a Gaussian distribution. Its probability density function (PDF) is defined as:

$$PDF(\text{Err}_t) = \frac{1}{\sqrt{2\pi\beta^2}} \exp\left(-\frac{(\text{Err}_t - \alpha)^2}{2\beta^2}\right) \tag{10}$$

where α is the mean, representing the central tendency of the error, and β is the standard deviation, reflecting the degree of error dispersion. The Gaussian distribution curve is symmetrically bell-shaped, with the mean at the center. The standard deviation controls the curve width, illustrating the extent of variation around the mean.

The goal of prediction interval is to determine a range within which future observations are likely to fall with high probability. To better capture the uncertainty in steam flow predictions, we propose a method for constructing prediction intervals based on the assumption that prediction errors follow a Gaussian distribution. This approach calculates the upper and lower bounds at different confidence levels. For instance, at a 95% confidence level, the prediction interval for a Gaussian distribution is given by:

$$[\alpha - 1.96\beta, \alpha + 1.96\beta] \tag{11}$$

This interval suggests that there is a 95% probability that future observations will fall within this range, offering a clear probabilistic basis for prediction interval. The method is simple, highly interpretable, and effectively handles noise and outliers due to the symmetric long-tail properties of the Gaussian distribution. By calculating prediction intervals, extreme values can be identified and excluded, preventing them from distorting the model. This is particularly useful for managing uncertainties and noise in industrial processes.

IV. Experimental design and framework

A. Data description

The dataset used in this study was collected from a waste treatment facility in Qingdao, Shandong Province. It records the operational parameters of the incineration system from 00:00 on July 4, 2024, to 00:00 on August 4, 2024, with data collected at a frequency of one measurement per second by multiple instruments. The dataset includes 105 distinct variables, such as the temperature at the inlet of the recirculating flue gas fan, the pressure in the superheated steam header, the outlet temperature of the superheated steam, and the concentrations of NO and CO gases, among other critical parameters.

B. Experimental framework

The basic experimental framework of the steam flow prediction model includes data preprocessing, data splitting, feature engineering, self-supervised pre-training and fine-tuning, and the construction of prediction intervals. The main steps are as follows:

Step 1: To ensure the integrity and validity of the data, the first step is to preprocess the relevant data from the waste incineration process. This process includes operations such as checking and filling missing values, as well as removing outliers. Since the data is collected at a frequency of once per second, using the raw data directly may lead to redundancy and increased computational complexity. To improve data processing efficiency and reduce the impact of ultra-short-term fluctuations on model predictions, the data is resampled using a 15-second moving average. This method smooths out data fluctuations while preserving important trend information.

Step 2: In this experiment, data from the 21 days after July 4 is used as the pre-training dataset, split into training and validation sets in an 8:2 ratio. Data from the 9 days after July 26 serves as the fine-tuning dataset, divided into training, validation, and test sets in an 8:1:1 ratio, as shown in Fig. 4. The dimensional details of each dataset are summarized in Table I.

TABLE I
NUMBER OF EACH SUBSET

Dataset name	Description	Samples	Features
Pretrain-Train	Training data of pre-training	96,640	105
Pretrain-Val	Validation data of pre-training	24,161	105
Finetune-Train	Training data of fine-tuning	41,313	105
Finetune-Val	Validation data of fine-tuning	5,025	105
Finetune-Test	Testing data of fine-tuning	5,026	105

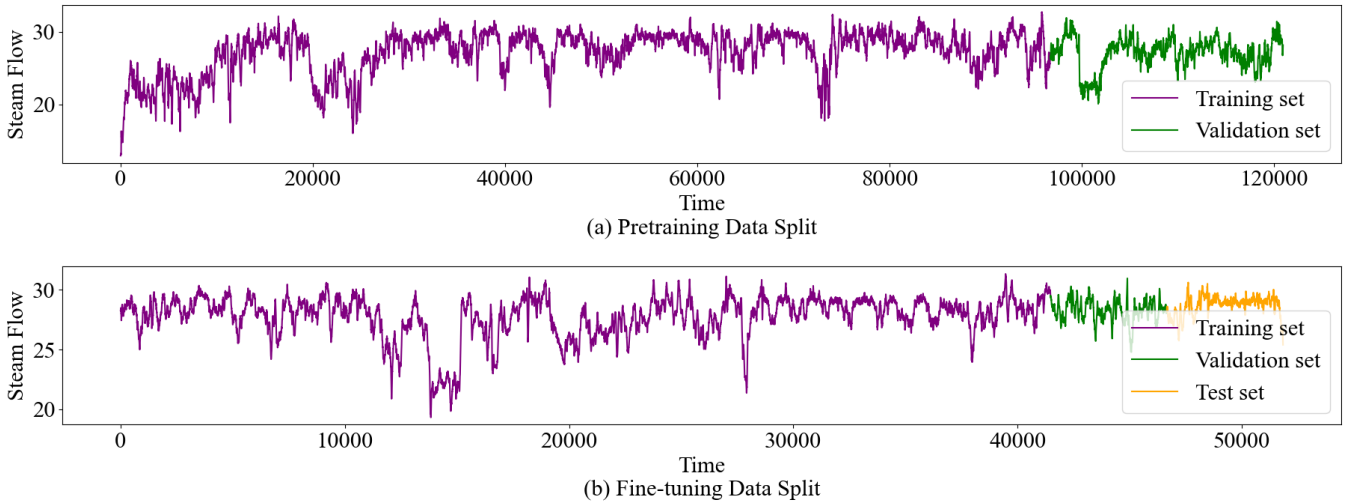

Fig. 4. The figure shows the data splits for steam flow in the pre-training and fine-tuning phases.

Step 3: We utilize the novel feature selection method (SEFS) on 80% of Pretrain-Train. Specifically, 70% of Pretrain-Train is used for feature selection, while 10% is reserved for validation to ensure the robustness of the extracted features. Through this innovative approach, we identify 8 highly discriminative and domain-relevant feature variables (as detailed in Table II). These features are rigorously selected based on their contribution to predictive accuracy and are directly integrated into the pre-training and fine-tuning stages to enhance model generalizability. Moreover, to capture the temporal dynamics of the system, we incorporate date information as an explicit temporal feature, enabling the model to better understand time-related dependencies and patterns.

TABLE II
FEATURE SELECTION RESULTS.

Input	Variable Name
x1	Drum Pressure 2
x2	Furnace 2 Steam Drum Pressure
x3	Steam Outlet Pressure
x4	Primary Desuperheating Flow
x5	Secondary Desuperheating Flow
x6	Flue 1 Lower-Middle Left Temp
x7	Flue 1 Middle Left Temp
x8	Flue 1 Lower Left Temp
x9 (target)	Steam Flow

Step 4: During the self-supervised pretraining stage, m features are selected from the Pre-training dataset for pretraining the GRU model using a 60% masking rate on target variables to foster learning of general data features. The model is then fine-tuned on the Finetune-Train and Finetune-Val datasets to enhance dataset-specific adaptability. It is worth noting that the Huber Loss [28] is used in both the pre-training and fine-tuning phases to balance small and large errors. Its mathematical expression is:

$$L_\delta(\hat{Q}, Q) = \begin{cases} \frac{1}{2}(\hat{Q} - Q)^2 & \text{if } |\hat{Q} - Q| \leq \delta \\ \delta \cdot (|\hat{Q} - Q| - 0.5\delta) & \text{if } |\hat{Q} - Q| > \delta \end{cases} \quad (12)$$

where $\hat{Q}$ and Q are the predicted and true steam flow values, respectively, and δ controls the transition between quadratic and linear loss, improving robustness to outliers.

Step 5: The prediction error of the Finetune-Val dataset is calculated, and the error is modeled using probability density estimation based on a Gaussian distribution to determine the upper and lower bounds at different confidence levels. These confidence intervals are then applied to the predicted values of the test set, ultimately constructing the prediction interval for steam flow. This approach ensures reasonable quantification of uncertainty.

The whole experimental flowchart of the proposed model is shown in Fig. 5, focusing on pre-training and fine-tuning phases. Initially, the model learns feature representations from historical data via a self-supervised method. It is then fine-tuned with data similar to the test set, improving its predictive accuracy and adaptability for the target period.

C. Evaluation metrics

In the analysis of point prediction results, several commonly used error evaluation metrics are employed to more accurately quantify the magnitude of prediction errors. These metrics include the Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE). The formulas for these metrics are as follows:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|Q_t - \hat{Q}_t| \quad (13)$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(Q_t - \hat{Q}_t)^2} \quad (14)$$

$$\text{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{Q_t - \hat{Q}_t}{Q_t}\right| \times 100\% \quad (15)$$

where N is the total number of predicted points, Q_t and $\hat{Q}_t$ represent the original steam flow value and predicted steam flow value at time t, respectively.

To assess the performance of prediction intervals, we adopt three evaluation metrics: the Prediction Interval Coverage

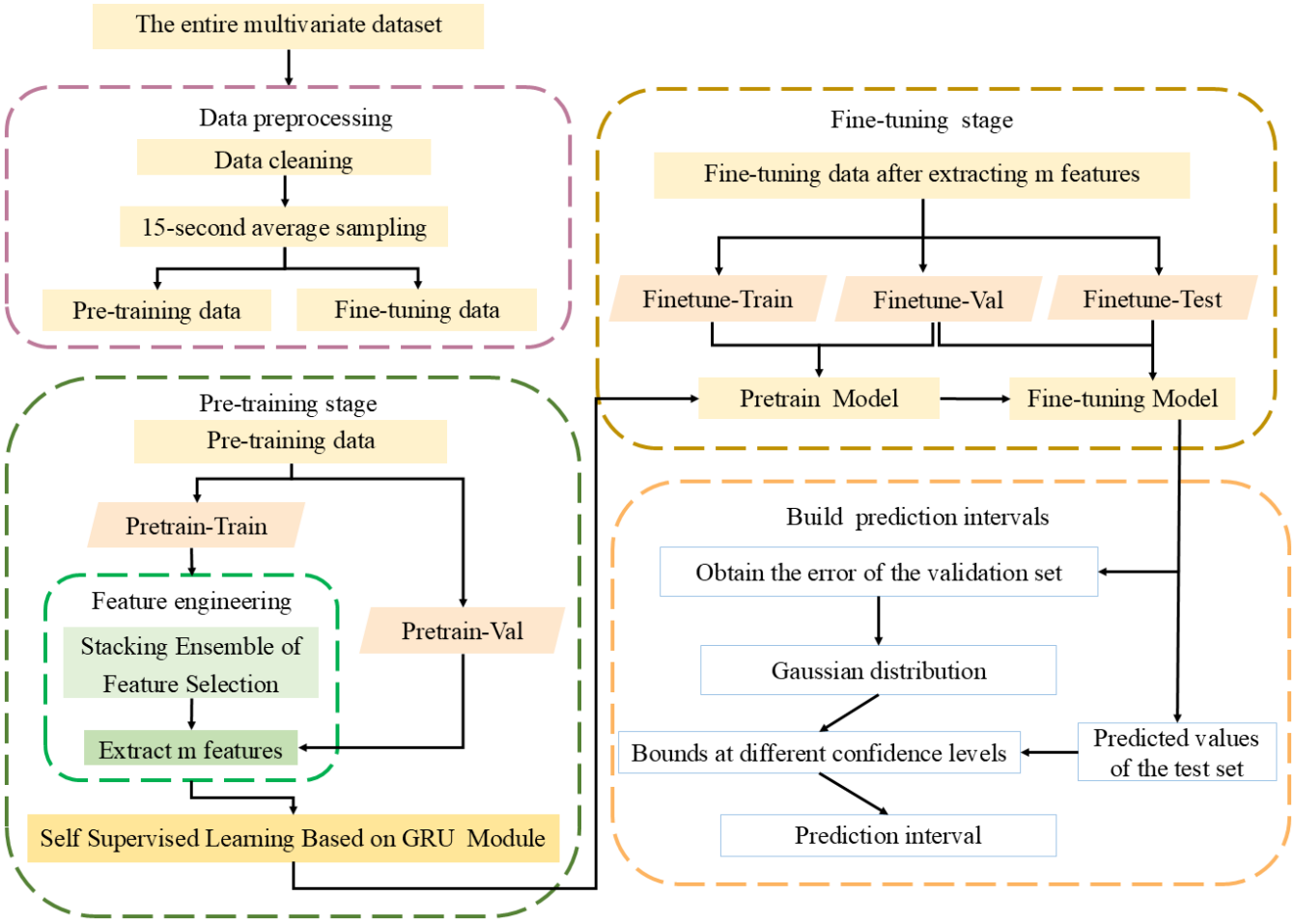

Fig. 5. The whole experiment process of the proposed model.

Probability (PICP), the Prediction Interval Normalized Average Width (PINAW), and a comprehensive index, the F-value. Their corresponding formulas are as follows:

$$\text{PICP} = \frac{1}{N}\sum_{j=1}^{N} f(\cdot)(\text{Down}_t \le Q_t \le \text{Up}_t) \tag{16}$$

$$\text{PINAW} = \frac{1}{N}\sum_{j=1}^{N} \frac{\text{Up}_t - \text{Down}_t}{Q_{\max} - Q_{\min}} \tag{17}$$

$$F = \lambda_1 \cdot \text{PICP} - \lambda_2 \cdot \text{PINAW} \tag{18}$$

where Down_t represents the lower bound of the prediction interval at time t, and Up_t denotes the upper bound of the prediction interval at time t. The symbol $f(\cdot)$ stands for the indicator function, which takes the value 1 if the condition $\text{Down}_t \le Q_t \le \text{Up}_t$ is satisfied, and 0 otherwise. $Q_{\max}$ and $Q_{\min}$ are the maximum and minimum values of the observed data, respectively. Finally, λ_1 and λ_2 are the weighting coefficients for the PICP and PINAW, balancing the trade-off between coverage and interval width in the comprehensive index F.

In these evaluation metrics, a higher PICP value is desirable, as it indicates that a larger proportion of true values fall within the predicted interval, representing better interval coverage. On the other hand, a lower PINAW value is preferred, as it suggests a narrower prediction interval, reflecting higher precision. The F-value serves as a comprehensive index, and a higher F-value indicates a better balance between interval coverage (PICP) and interval width (PINAW). Therefore, the optimal model would achieve a high PICP, a low PINAW, and consequently, a high F-value.

V. Results Analysis and Comparison

In this section, we assess the predictive performance of the proposed method from two aspects: point prediction and prediction interval.

A. Comparison of point forecasting results

In this experiment, we use historical data from the past 30 minutes to predict steam flow values for the next 3, 5, and 10 minutes. Fig. 6 illustrates the point prediction results on the test set for the three forecasting horizons. To evaluate the effectiveness of the proposed SEFS-GSSL model in point forecasting, we compare it with eight time series models, namely: Linear, LSTM, WaveNet [29], Transformer, TCN [30], DSANet [31], N-BEATS [32], and TFT [33]. The prediction errors of each model on the fine-tuned dataset are summarized in Table III.

From these prediction results, the SEFS-GSSL model demonstrates strong performance across all prediction tasks. Specifically, in the 3-minute and 5-minute forecasting tasks, SEFS-GSSL consistently outperformes all baseline models across key error metrics—MAE, RMSE, and MAPE. In the 3-minute prediction task, SEFS-GSSL

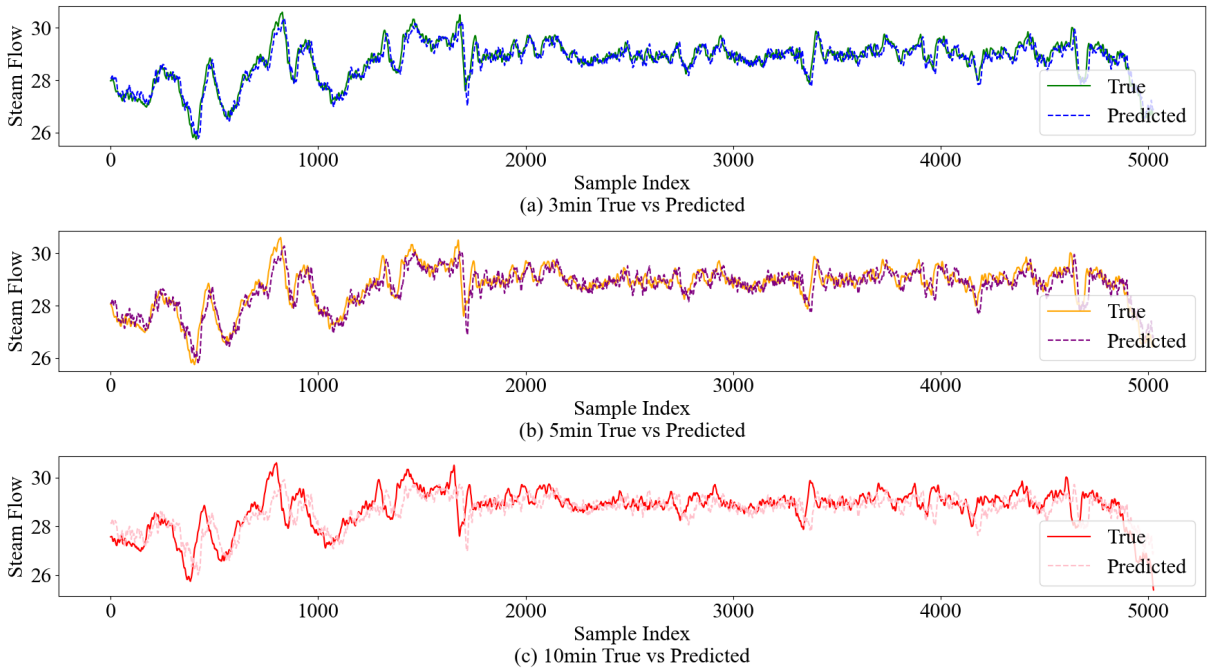

Fig. 6. The figure shows that prediction errors increase with forecast time, indicating higher accuracy for ultra-short-term and greater uncertainty for longer-term steam flow forecasts.

TABLE III
COMPARISON OF POINT PREDICTION RESULTS FOR THREE TIME PERIODS.

Models	3min			5min			10min		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
Linear	0.348	0.424	1.206%	0.398	0.485	1.383%	0.506	0.617	1.761%
LSTM	0.306	0.397	1.060%	0.360	0.465	1.247%	0.468	0.612	1.626%
WaveNet	0.264	0.339	0.913%	0.321	0.413	1.114%	0.434	**0.564**	1.509%
Transformer	0.260	0.328	0.905%	0.311	0.402	1.083%	0.439	0.571	1.526%
TCN	0.262	0.337	0.908%	0.318	0.412	1.100%	0.476	0.614	1.649%
DSANet	0.217	0.279	0.755%	0.309	0.402	1.073%	0.592	0.760	2.049%
N-Beats	0.310	0.400	1.075%	0.378	0.488	1.312%	0.497	0.635	1.727%
TFT	0.310	0.392	1.080%	0.335	0.432	1.163%	0.463	0.592	1.610%
SEFS-GSSL	**0.204**	**0.266**	**0.710%**	**0.272**	**0.365**	**0.949%**	**0.422**	0.565	**1.474%**

TABLE IV
THREE TIME PERIODS OF ABLATION EXPERIMENTS.

Models	3min			5min			10min		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
GRU	0.262	0.340	0.908%	0.345	0.449	1.195%	0.458	0.606	1.592%
GRU-SEFS	0.217	0.280	0.752%	0.292	0.389	1.015%	0.438	0.581	1.529%
GRU-SSL	0.235	0.305	0.814%	0.292	0.386	1.013%	0.429	0.579	1.493%
GRU-SSL-RF	0.209	0.270	0.727%	0.278	0.371	0.969%	0.428	**0.565**	1.494%
SEFS-GSSL	**0.204**	**0.266**	**0.710%**	**0.272**	**0.365**	**0.949%**	**0.422**	**0.565**	**1.474%**

achieved MAE, RMSE, and MAPE values of 0.204, 0.266, and 0.710%, respectively. These values were significantly lower than those of the other models. For instance, the next best model, DSANet, reported corresponding values of 0.217, 0.279, and 0.755%, respectively. SEFS-GSSL improved MAE by 6.0% (from 0.217 to 0.204), RMSE by 4.7% (from 0.279 to 0.266), and MAPE by 5.9% (from 0.755% to 0.710%) for the 3-minute prediction. These improvements are particularly important in real-world applications where accurate ultra-short-term predictions are critical for operational decision-making.

In the 5-minute prediction task, SEFS-GSSL continued to outperform the baseline models, achieving MAE, RMSE, and MAPE values of 0.272, 0.365, and 0.949%, respectively. These values were again significantly lower than those of the next best model, DSANet, whose corresponding values were 0.309, 0.402, and 1.073%. SEFS-GSSL achieved a 12.0% improvement in MAE (from 0.309 to 0.272), a 9.2% improvement in RMSE (from 0.402 to 0.365), and an 11.5% improvement in MAPE (from 1.073% to 0.949%). This shows that, even with slightly longer-term predictions, where models often struggle to maintain stability due to the accumulation of uncertainties, SEFS-GSSL still maintains high accuracy. The improvement in RMSE is

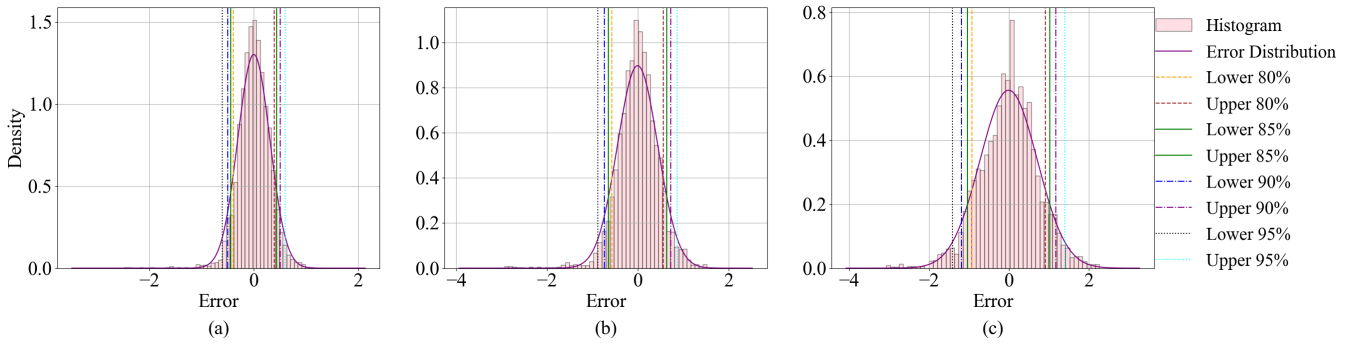

Fig. 7. The figure displays the Gaussian distribution of prediction errors for steam flow at three forecast times: (a) 3 min, (b) 5 min, and (c) 10 min. As the forecast time increases, the error distribution widens, indicating greater variance in prediction accuracy over longer intervals. Each plot includes confidence intervals at 80%, 85%, 90%, and 95%, shown with vertical lines to illustrate error bounds.

TABLE V
UPPER AND LOWER BOUNDS OF PREDICTION ERRORS AT DIFFERENT CONFIDENCE LEVELS.

Confidence Levels	3min		5min		10min	
	Lower	Upper	Lower	Upper	Lower	Upper
80%	-0.3937	0.3918	-0.5738	0.5665	-0.9254	0.9131
85%	-0.4422	0.4402	-0.6441	0.6368	-1.0387	1.0265
90%	-0.5051	0.5032	-0.7354	0.7281	-1.1860	1.1737
95%	-0.6017	0.5998	-0.8756	0.8683	-1.4121	1.3998

particularly significant, as it indicates that SEFS-GSSL reduces prediction variability, making the model more stable and reliable over extended periods.

Although the WaveNet model achieved the lowest RMSE (0.564) in the 10-minute task, SEFS-GSSL maintained an advantage in both MAE and MAPE. Specifically, SEFS-GSSL's MAE of 0.422 was 2.76% better than WaveNet's MAE of 0.434. Similarly, SEFS-GSSL achieved a MAPE of 1.474%, which was slightly better than WaveNet's MAPE of 1.509%. While the RMSE metric for the 10-minute prediction was lower in WaveNet, the fact that SEFS-GSSL achieves better MAE and MAPE indicates that it is more reliable in terms of these metrics, which is particularly important in real-world scenarios where consistent and stable predictions are critical for decision-making processes.

These results confirm that SEFS-GSSL consistently delivers both high accuracy and stable performance across ultra-short-term horizons, establishing it as a valuable tool for real-time industrial forecasting applications.

B. Ablation study

In the point prediction task, to evaluate the impact of two key components in the SEFS-GSSL model—the stacking ensemble feature selection method and the SSL mechanism—on the overall model performance, we conducted ablation experiments. Specifically, five scenarios were investigated: 1) GRU, using only the GRU model; 2) GRU-SEFS, using GRU and the stacking ensemble feature selection method; 3) GRU-SSL, using GRU and the SSL mechanism; 4) GRU-SSL-RF, using GRU, the SSL mechanism, and the best-performing feature selection method among the three, namely Random Forest; 5) SEFS-GSSL, using both the stacking ensemble feature selection method and the SSL mechanism.

The results presented in Table IV clearly demonstrate the individual contributions and synergistic effects of the model components. Firstly, the comparison between GRU and GRU-SEFS shows that the introduction of the stacking ensemble feature selection significantly enhances prediction accuracy across all three forecasting horizons (3 min, 5 min, and 10 min). For example, in the 3-minute prediction task, MAE decreased from 0.262 to 0.217, RMSE from 0.340 to 0.280, and MAPE from 0.908% to 0.752%, indicating that SEFS effectively selected and integrated the most informative features, thereby improving the model's ability to capture the temporal dynamics of steam flow.

Secondly, the standalone application of the SSL mechanism (GRU-SSL compared to GRU) also yields consistent performance improvements. The mask-based SSL pretraining enables the model to learn robust and generalized feature representations, enhancing its capability to handle data variability and noise. In the 3-minute prediction task, MAE was reduced from 0.262 to 0.235, RMSE from 0.340 to 0.305, and MAPE from 0.908% to 0.814%. Similar improvements were observed in the 5-minute and 10-minute prediction tasks, further validating the SSL mechanism's effectiveness in enhancing model generalization and prediction stability.

Thirdly, a comparison between GRU-SSL-RF and GRU-SEFS reveals that although GRU-SSL-RF integrates SSL with a single feature selection method—RF—its overall performance is still inferior to that of GRU-SEFS. This indicates that a single feature selection method is less effective than the proposed stacking ensemble feature selection strategy in improving model performance. Taking the 3-minute prediction task as an example, GRU-SSL-RF achieved MAE, RMSE, and MAPE of 0.209, 0.270, and 0.727%, respectively, whereas the corresponding metrics for GRU-SEFS were 0.217, 0.280, and 0.752%, which further confirms the superiority of the ensemble strategy in aggregating complementary features.

In summary, the ablation experiments clearly confirm

TABLE VI
INDICATORS FOR 3-MINUTE PREDICTION INTERVAL.

Models	80%			85%			90%			95%		
	PICP	PINAW	F value	PICP	PINAW	F value	PICP	PINAW	F value	PICP	PINAW	F value
SEFS-GSSL	**0.876**	**0.1618**	**1.7145**	0.908	**0.1817**	1.7266	0.938	**0.2076**	1.7303	**0.969**	**0.2474**	**1.7212**
GRU-SSL-RF	0.873	0.1625	1.7105	**0.911**	0.1825	**1.7285**	**0.941**	0.2086	**1.7324**	0.967	0.2485	1.7185
GRU-SEFS	0.874	0.1664	1.7076	**0.911**	0.1869	1.7241	0.936	0.2136	1.7224	0.964	0.2545	1.7095
GRU-SSL	0.862	0.1831	1.6789	0.899	0.2057	1.6933	0.932	0.2350	1.6970	0.966	0.2800	1.6860
GRU	0.838	0.1813	1.6567	0.870	0.2036	1.6664	0.909	0.2327	1.6763	0.947	0.2772	1.6698

TABLE VII
INDICATORS FOR 5-MINUTE PREDICTION INTERVAL.

Models	80%			85%			90%			95%		
	PICP	PINAW	F value	PICP	PINAW	F value	PICP	PINAW	F value	PICP	PINAW	F value
SEFS-GSSL	0.901	**0.2348**	**1.6665**	0.923	**0.2637**	**1.6589**	0.941	**0.3013**	**1.6394**	0.968	**0.3591**	**1.6091**
GRU-SSL-RF	0.901	0.2394	1.6616	0.925	0.2689	1.6561	0.943	0.3073	1.6357	**0.970**	0.3662	1.6038
GRU-SEFS	**0.904**	0.2479	1.6561	0.928	0.2785	1.6495	**0.947**	0.3182	1.6288	0.969	0.3792	1.5898
GRU-SSL	0.897	0.2461	1.6509	0.923	0.2764	1.6466	**0.947**	0.3159	1.6311	0.968	0.3764	1.5916
GRU	0.837	0.2447	1.5923	0.876	0.2749	1.6011	0.909	0.3141	1.5949	0.949	0.3743	1.5747

TABLE VIII
INDICATORS FOR 10-MINUTE PREDICTION INTERVAL.

Models	80%			85%			90%			95%		
	PICP	PINAW	F value	PICP	PINAW	F value	PICP	PINAW	F value	PICP	PINAW	F value
SEFS-GSSL	0.885	**0.3528**	1.5326	0.916	0.3963	**1.5197**	0.945	0.4529	**1.4918**	0.975	0.5396	**1.4353**
GRU-SSL-RF	0.895	0.3603	**1.5347**	0.922	0.4048	1.5172	0.949	0.4625	1.4865	**0.979**	0.5511	1.4279
GRU-SEFS	**0.904**	0.3761	1.5279	**0.932**	0.4225	1.5095	**0.957**	0.4828	1.4742	0.978	0.5752	1.4028
GRU-SSL	0.903	0.3686	1.5344	0.927	0.4140	1.5130	0.951	0.4731	1.4779	0.977	0.5637	1.4133
GRU	0.866	0.3449	1.5211	0.897	**0.3874**	1.5096	0.926	**0.4427**	1.4833	0.955	**0.5275**	1.4275

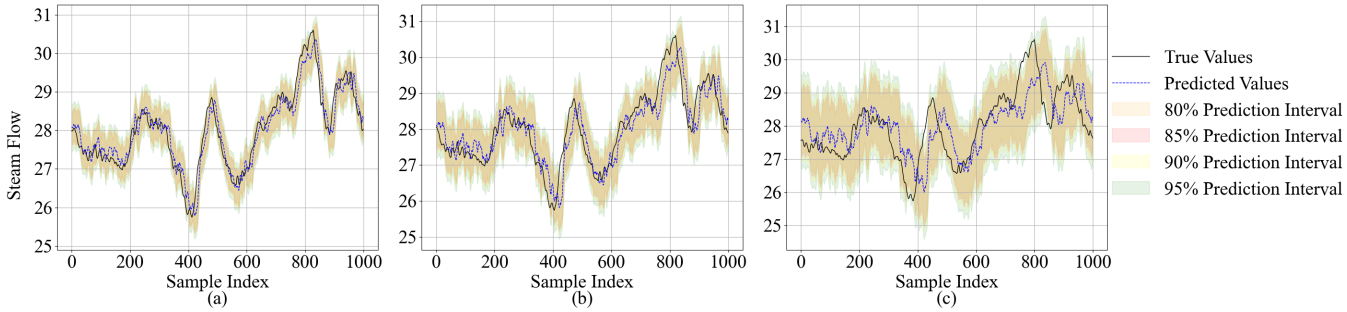

Fig. 8. The figure shows prediction intervals for the first 1000 samples of steam flow at different time horizons: (a) 3 min, (b) 5 min, and (c) 10 min. Prediction intervals widen as the forecast horizon increases, indicating higher uncertainty over longer periods. The blue dashed line represents predicted values, the black solid line denotes true values, and shaded areas indicate various confidence intervals.

that both the stacking ensemble feature selection and the SSL mechanisms are crucial contributors to the performance enhancements of the SEFS-GSSL model. The former enriches the input quality through a diverse and comprehensive feature set, while the latter improves prediction accuracy and stability by enhancing representation learning. The synergy between these two components results in a more robust and precise steam flow forecasting model, demonstrating the value of advanced feature selection techniques combined with SSL in complex industrial time series prediction tasks.

C. The results of prediction interval

In this section, we analyze the prediction interval results to assess the uncertainty quantification capability of the proposed SEFS-GSSL model. First, we estimate the prediction errors for three time intervals on the validation set using a Gaussian distribution and calculate the upper and lower bounds at different confidence levels, as shown in Fig. 7. The values of these bounds are summarized in Table V. Finally, we apply the calculated bounds to the predicted values on the test set, obtain the prediction intervals for steam flow, and compare them with the ablation models. The results are summarized in Table VI, VII, and VIII. As shown in Fig. 8, the prediction intervals for the three forecast periods on the test set are displayed for 1,000 data points, respectively.

From these results, it is evident that the SEFS-GSSL model consistently outperforms other ablation models on all prediction tasks (3 min, 5 min, and 10 min) and

confidence levels, exhibiting particularly strong performance in terms of PICP and PINAW. For instance, in the 3-minute prediction task at the 80% confidence level, SEFS-GSSL achieves a PICP of 0.876, surpassing the baseline model GRU-SSL-RF, which attains 0.873. This indicates superior interval coverage. Moreover, the PINAW of SEFS-GSSL is 0.1618, narrower than that of other models, demonstrating its ability to generate more precise prediction intervals.

As the confidence level increases to 90% and 95%, SEFS-GSSL maintains a high level of interval coverage. At the 90% confidence level, its PICP is 0.938, slightly less than that of GRU-SSL-RF (0.941), yet PINAW remains compact at 0.2076, indicating that the model sustains high precision with narrower intervals even under stringent confidence requirements. At the 95% confidence level, SEFS-GSSL attains a PICP of 0.969, exceeding all other ablation models, with a PINAW of just 0.2474, which signifies its capability to produce relatively tight intervals even at very high confidence levels, thereby providing greater assurance in uncertainty quantification for industrial applications.

In the 5-minute and 10-minute prediction tasks, SEFS-GSSL similarly demonstrates robust uncertainty quantification capabilities. Notably, in the 10-minute prediction task, SEFS-GSSL achieves the highest F-values across all confidence levels. For example, at 95% confidence, the F-value of SEFS-GSSL reaches 1.4353, outperforming the baseline GRU model's 1.4275. This indicates that SEFS-GSSL not only provides high-coverage prediction intervals but also maintains compact interval widths, offering more precise and reliable forecasts in complex industrial environments.

Overall, the SEFS-GSSL model exhibits outstanding uncertainty quantification performance across all prediction tasks, particularly excelling in PICP and PINAW metrics compared to baseline models, highlighting its advantages in ultra-short-term forecasting scenarios. In practical applications, SEFS-GSSL delivers high-quality prediction intervals that simultaneously increase coverage rates and maintain narrow widths, making it especially valuable in complex industrial settings.

VI. Conclusions and future works

This paper proposes a time series prediction model, SEFS-GSSL, which integrates a stacked ensemble-based feature selection strategy with GRU-based SSL. Its effectiveness is validated through extensive experiments on steam flow forecasting. The SEFS-GSSL model consistently outperforms baseline and ablation models in both point forecasting and interval prediction tasks across multiple forecasting horizons, achieving significant gains across multiple error metrics. Specifically, by leveraging ensemble-based feature selection and SSL, SEFS-GSSL effectively models complex industrial data, leading to substantial improvements in accuracy and generalization. Additionally, the SEFS-GSSL model exhibits strong uncertainty quantification capabilities, generating tighter and more comprehensive prediction intervals, thus providing reliable support for risk assessment and decision-making in industrial applications.

Despite these promising results, there remains room for further improvement. Future work may explore: (1) enhancing the SSL mechanism for improved adaptability to non-stationary industrial environments; and (2) tailoring the model to domain-specific requirements to boost practical applicability and deployment efficiency.

References

[1] T. Wang, J. Tang, L. Aljerf, J. Qiao, and M. Alajlani, "Emission reduction optimization of multiple flue gas pollutants in municipal solid waste incineration power plant," Fuel, vol. 381, Part A, p. 133382, 2025.

[2] S. Zheng, J. Zhang, H. Qu, W. Cai, Y. Wang, and Q. Lu, "In-situ measurement of alkali metal concentrations and temperatures during the incineration of municipal solid waste and sludge," Fuel, vol. 373, p. 132306, 2024.

[3] L. Chen, C. Wang, R. Zhong, et al., "Prediction of main parameters of steam in waste incinerators based on BAS-SVM," Sustainability, vol. 15, no. 2, p. 1132, 2023.

[4] D. M. N. Bristol, I. H. V. Gue, and A. T. Ubando, "A state-of-the-art review on machine learning based municipal waste to energy system," Cleaner Energy Systems, vol. 9, p. 100143, 2024.

[5] Y. Wei, Y. Xue, J. Yin, and W. Ni, "Prediction of municipal solid waste generation in China by multiple linear regression method," International Journal of Computer Applications, vol. 35, no. 3, pp. 136-140, 2015.

[6] Y. Wang, Z. Tang, and B. Zhao, "Modeling of boiler steam flow based on adaptive least squares support vector machine," in Proceedings of the 9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), Hangzhou, China, pp. 187-190, 2017.

[7] Z. Fu, M. Qi, and Y. Jing, "Regression forecast of main steam flow based on mean impact value and support vector regression," in Proceedings of the Asia-Pacific Power and Energy Engineering Conference, IEEE, 2012.

[8] S. Golgiyaz, M. F. Talu, and C. Onat, et al., "Artificial neural network regression model to predict flue gas temperature and emissions with the spectral norm of flame image," Fuel, vol. 255, p. 115827, 2019.

[9] W. Lu, W. Huo, and H. Gulina, et al., "Development of machine learning multi-city model for municipal solid waste generation prediction," Frontiers of Environmental Science and Engineering, vol. 16, no. 9, 2022.

[10] K. Xu, "Industrial steam quantity prediction model based on grid search-optimized support vector regression," in Proceedings of the IEEE 3rd International Conference on Electrical Engineering, Big Data Algorithms (EEBDA), Changchun, China, pp. 1088-1094, 2024.

[11] Y. Xue and X. Liu, "Detoxification, solidification and recycling of municipal solid waste incineration fly ash: A review," Chemical Engineering Journal, vol. 420, p. 130349, 2021.

[12] Qian Li, and Yuan Cao, "Short-Term Electricity Load Forecasting Based on GWO-VMD-ISSA-LSTM Model," IAENG International Journal of Applied Mathematics, vol. 55, no. 5, pp1213-1226, 2025.

[13] Guangming Zhu, Dong Liang, Yunyan Xu, Shuting Wu, and Wenbin Qin, "A Line Loss Prediction Method of Distribution Network Based on Improved Grey Correlation Analysis and SSA-GRU Neural Network," IAENG International Journal of Applied Mathematics, vol. 55, no. 5, pp1105-1115, 2025.

[14] A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention is all you need," in Advances in Neural Information Processing Systems, vol. 30, 2017.

[15] L. Tian, K. Lin, Y. Zhao, et al., "Combustion performance of fine screenings from municipal solid waste: Thermo-kinetic investigation and deep learning modeling via TG-FTIR," Energy, vol. 243, p. 122783, 2022.

[16] X. Hu, W. Wang, J. Tang, M. Liu, "Multi-step ahead prediction of main steam flow using PCA and Pyraformer in MSWI process," in Proceedings of the 35th Chinese Control Decision Conference (CCDC), 2023, pp. 372–376.

[17] H. Wu, Y. Mu, Y. Yang, et al., "Pyraformer: Low-complexity pyramidal attention for long-range time series modeling," in Proceedings of the 38th International Conference on Machine Learning (ICML), 2021.

[18] J. J. Quiñones, L. R. Pineda, J. Ostanek, L. Castillo, "Towards smart energy management for community microgrids: Leveraging deep learning in probabilistic forecasting of renewable energy sources," Energy Conversion and Management, vol. 293, p. 117440, 2023.

[19] S. Konstantakos, J. Cani, I. Mademlis, D. I. Chalkiadaki, Y. M. Asano, E. Gavves, G. Th. Papadopoulos, "Self-supervised visual learning in the low-data regime: A comparative evaluation," Neurocomputing, vol. 620, p. 129199, 2025.

[20] Z. Ying, D. Cheng, C. Chen, X. Li, P. Zhu, Y. Luo, Y. Liang, "Predicting stock market trends with self-supervised learning," Neurocomputing, vol. 568, p. 127033, 2024.

[21] Amalia Amalia, Maya Silvi Lydia, Muhammad Anggia Muchtar, Fuzy Yustika Manik, Sinu, and Dani Gunawan, "Mitigating Bias and Assessment Inconsistencies with BERT-Based Automated Short Answer Grading for the Indonesian Language," IAENG International Journal of Computer Science, vol. 52, no. 3, pp533-545, 2025.
[22] K. Zhang, Q. Wen, C. Zhang, et al., "Self-Supervised Learning for Time Series Analysis: Taxonomy, Progress, and Prospects," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 46, no. 10, pp. 6775–6794, Oct. 2024.
[23] F. Alobaid, W. A. K. Al-Maliki, T. Lanz, et al., "Dynamic simulation of a municipal solid waste incinerator," Energy, vol. 149, pp. 230–249, 2018.
[24] K. Myroniuk, O. Voznyak, Y. Yurkevych, et al., "Technical and economic efficiency after the boiler room renewal," in Proceedings of the International Scientific Conference on EcoComfort and Current Issues of Civil Engineering, Cham, Switzerland: Springer International Publishing, 2020, pp. 311–318.
[25] E. Pan, X. Mei, Q. Wang, Y. Ma, J. Ma, "Spectral-spatial classification for hyperspectral image based on a single GRU," Neurocomputing, vol. 387, pp. 150–160, 2020.
[26] L. Gao, X. Wang, J. Song, Y. Liu, "Fused GRU with semantic-temporal attention for video captioning," Neurocomputing, vol. 395, pp. 222–228, 2020.
[27] W. Zhang, Y. He, S. Yang, "A multi-step probability density prediction model based on Gaussian approximation of quantiles for offshore wind power," Renewable Energy, vol. 202, pp. 992–1011, 2023.
[28] S. Huang, Z. Zeng, S. Jiang, "Generalization bounds for pairwise learning with the Huber loss," Neurocomputing, vol. 622, p. 129265, 2025.
[29] A. van den Oord, S. Dieleman, H. Zen, et al., "WaveNet: A generative model for raw audio," in Proceedings of the 9th ISCA Speech Synthesis Workshop, 2016, pp. 125–125.
[30] S. Bai, J. Z. Kolter, V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," arXiv preprint, arXiv:1803.01271, 2018.
[31] Z. Huang, S. Zhong, Z. Yue, "DSANet: Dual self-attention network for multivariate time series forecasting," arXiv preprint, arXiv:2003.01006, 2020.
[32] B. N. Oreshkin, D. Carpov, N. Chapados, et al., "N-BEATS: Neural basis expansion analysis for time series forecasting," in Proceedings of the International Conference on Learning Representations (ICLR), 2020.
[33] B. Lim, S. Ö. Arık, N. Loeff, et al., "Temporal fusion transformers for interpretable multi-horizon time series forecasting," International Journal of Forecasting, vol. 37, no. 4, pp. 1748–1764, 2021.