China Instrument and Control Society
Instrumentation
INSTRUMENTATION, Vol. 12, No. 2, June 2025
https://doi.org/10.15878/j.instr.202500277

Article

CNN-ALSTM Soft-Sensing Model for Resistivity Gradient in CZ Monocrystalline Silicon Wafers

Zhiheng Zhang, Zengguo Tian*

School of Physics and Microelectronics, Zhengzhou University, Zhengzhou, Henan 450000, China

* Corresponding author email: tianzg@zzu.edu.cn

Abstract: This study addresses the challenge of real-time resistivity gradient measurement in the Czochralski (CZ) silicon production process. Due to the inability to directly measure this parameter, we propose a Long Short-Term Memory soft-sensing model based on Convolutional Neural Network (CNN) and attention mechanism (CNN-ALSTM) that enhances traditional LSTM by integrating CNN and attention mechanism to overcome time lag variations during silicon pulling. The CNN module extracts spatial features from multi-source sensor data, while the attention-enhanced LSTM (ALSTM) dynamically adjusts historical parameter weights, enabling accurate resistivity gradient prediction. Experiments with real production data show that CNN-ALSTM outperforms SVR, FNN, RNN, XGBoost, and GRU, improving prediction accuracy by 11.76%, 16.67%, 21.05%, 30.23%, and 9.09%, respectively. This soft-sensing approach enhances real-time monitoring and optimization of monocrystalline silicon growth.

Keywords: CZ monocrystalline silicon; soft-sensing model; CNN-ALSTM; resistivity gradient

Citation: Zhiheng Zhang, Zengguo Tian."CNN-ALSTM Soft-Sensing Model for Resistivity Gradient in CZ Monocrystalline Silicon Wafers. " Instrumentation 12, no.2 (June 2025). https://doi.org/10.15878/j.instr.202500277

1 Introduction

Monocrystalline silicon wafers, the cornerstone of the semiconductor industry, directly influence integrated circuit performance and reliability, with over 95% of semiconductor devices relying on silicon substrates[1]. The Czochralski (CZ) method, dominant in producing high-quality monocrystalline silicon[2], faces challenges from complex phenomena—melt thermal fluctuations, impurity partitioning, and solid-liquid interface dynamics—that create uneven radial resistivity gradients. These gradients degrade wafer electrical uniformity, causing defects like threshold voltage drift and leakage current, ultimately reducing chip yield.

Current industrial resistivity testing depends on offline four-point probe measurements, incurring a 6–8-hour lag that prevents real-time process adjustments. This limitation underscores the need for online soft-sensing to monitor resistivity gradients dynamically, a critical step for quality control in CZ silicon growth.

In industrial production, data-driven soft-sensor techniques have been extensively studied and deployed to address challenges in sensing critical parameters that are inherently difficult to measure directly. For instance, weld defect[3], volumetric flow[4] and initial boiling point of the diesel oil product[5] can now be accurately monitored through advanced soft-sensor models. Furthermore, these methods exhibit high prediction accuracy for product quality metrics, such as the final quality of red wine[6] and injection-molded products[7]. With the rapid advancement of deep learning technologies, key performance attributes of soft sensors—including adversarial robustness and interpretability—have been progressively enhanced[8,9].

In the CZ silicon production industry, the application of data-driven techniques and soft-sensor techniques are also exist—e. g., Liu et al. [10,11] modeled heat-mass transfer under magnetic fields—intelligent detection has also progressed. Examples include neural networks for oxygen content prediction[12,13] and defect detection[14-16], alongside diameter and pulling speed models[1,17,18], V/G value soft measurement[19,20], and process parameter prediction with optimization techniques[21-23]. Yet, these

studies based on data-driven techniques about CZ silicon production lack investigation into resistivity gradient measurement and neglect dynamic time-lag adjustments.

To address this gap, we propose a CNN-ALSTM soft-sensing model, enhancing traditional LSTM with convolutional neural networks (CNN) and attention mechanisms. This approach: 1) extracts spatial features from multi-source sensor data, reducing noise; 2) uses attention to dynamically adjust time lag values, improving responsiveness; and 3) maps production data directly to resistivity gradients. This lays a solid foundation for real-time optimization and adjustment of process parameters in the monocrystalline silicon pulling process, improving resistivity uniformity in the wafers.

2 Methods

2.1 CNN

The CNN is a data preprocessing layer specifically designed to extract valuable features and filter out noise [24]. The extracted key information is then input into the subsequent fully connected layer for classification or regression tasks. The CNN processes the input data and generates a new feature map. Originally designed for image feature extraction, the technique requires the input data to be in a structured matrix form [25]. Moreover, when using a CNN for multivariate time series data, it can consider the correlations between variables and efficiently extract features [26].

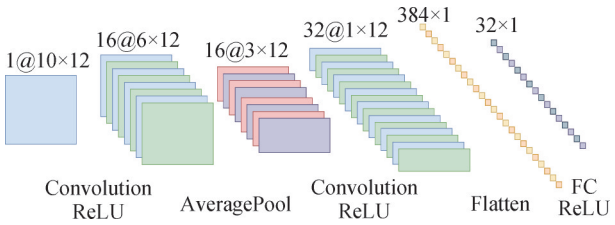

Fig.1 CNN Module Structure

This study presents a CNN tailored to the data of the monocrystalline silicon pulling process. The CNN architecture consists of an input matrix sized 10×12, which is processed through two convolutional layers, a pooling layer, and a fully connected layer, ultimately producing one-dimensional output data containing 32 values. The specific structure is illustrated in Fig. 1, and the CNN parameters are configured according to the specifications outlined in Table 1.

2.2 ALSTM

2.2.1 LSTM

To understand the complexities of ALSTM, it is essential to first comprehend the structural and functional nuances of Long Short-Term Memory (LSTM) networks. As a specialized variant of recurrent neural network (RNN), LSTMs can determine when to store, update, and retrieve information during training, thereby overcoming the limitations of conventional RNNs in handling long sequential tasks. Central to LSTM is the incorporation of gating units that adaptively control information flow, as illustrated in Fig. 2. The structure consists of four main components: the forgetting gate, the input gate, the cell state, and the output gate.

Table 1 CNN Module Structure Parameters

Type	#Filter	Kernel size	Stride	In-Out-Feature
Convolution	16	(5,1)	1	-
Activation(ReLU)	-	-	-	-
Pooling	-	(2,1)	(2,1)	-
Convolution	32	(3,1)	1	-
Activation(ReLU)	-	-	-	-
Flatten	-	-	-	-
Dense(32)	-	-	-	(384,32)

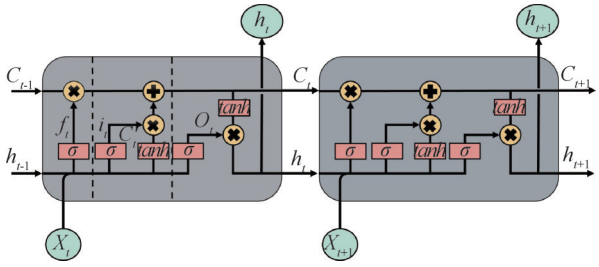

Fig.2 LSTM Module Structure

The forgetting gate determines which information is discarded from the cell state:

$$f_t=\sigma\left(W_f\times\left[h_{t-1},x_t\right]+b_f\right) \tag{1}$$

The input gate determines which new information is added to the cell state:

$$C'_t=\tanh\left(W_c\times\left[h_{t-1},x_t\right]+b_c\right) \tag{2}$$

$$i_t=\sigma\left(W_i\times\left[h_{t-1},x_t\right]+b_i\right) \tag{3}$$

The cell state is updated by integrating a portion of the previous cell state with the current candidate values, resulting in a new cell state:

$$C_t=f_t\times C_{t-1}+i_t\times C'_t \tag{4}$$

The output gate determines which components of the cell state are transmitted:

$$o_t=\sigma\left(W_o\times\left[h_{t-1},x_t\right]+b_o\right) \tag{5}$$

$$h_t=o_t\times\tanh\left(C_t\right) \tag{6}$$

2.2.2 Attention Mechanism

The term "attention" describes the human tendency to selectively focus on specific information while ignoring other perceptible data at any given time and place. This mechanism facilitates more efficient

information processing, leading to the development of attention mechanisms in computing [27]. The fundamental concept of an attention mechanism is to calculate the degree of similarity between two elements, e_i and e_j. One of the most prevalent approaches for measuring similarity is as follows [28]:

$$\text{score}\left(e_i, e_j\right) = v_a^T\left(W_a\left[e_i; e_j\right]\right) \tag{7}$$

To convert the model-generated similarity ratings into a probability distribution, a Softmax function is typically employed. It maps the raw similarity ratings to the [0, 1] range and ensures that their weighted sums equal 1.

$$\alpha_{ij} = \frac{\exp\left(\text{score}\left(e_i, e_j\right)\right)}{\sum_{k=1}^{n}\exp\left(\text{score}\left(e_i, e_j\right)\right)} \tag{8}$$

2.2.3 ALSTM

The ALSTM is an extension of the basic LSTM that integrates an attention mechanism. This mechanism evaluates the significance of various historical time periods and assigns different weights to production data from these periods. Consequently, it enables dynamic adjustment of time lags and enhances the model's capacity to process time series data. The structure of the ALSTM module is illustrated in Fig.3.

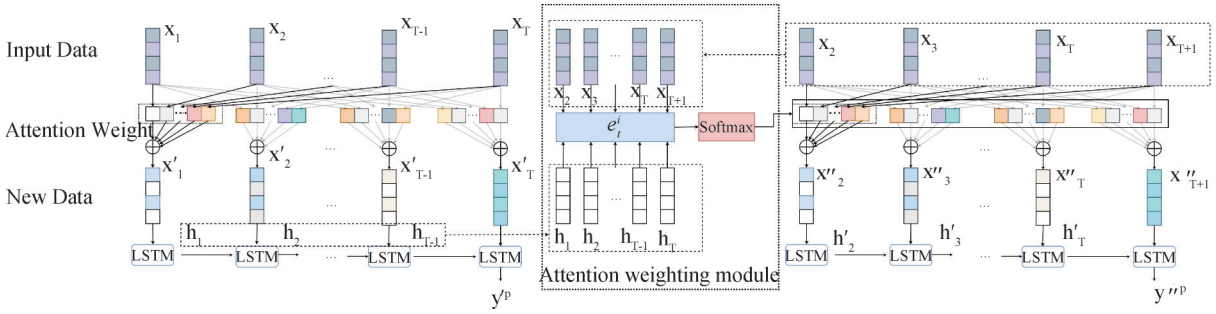

Fig.3 ALSTM Module Structure

Data x_1 to x_T from different time periods output by the convolutional layer are input into the attention module to prevent equal weighting of each time period. Instead, the attention value α_t^i for each time period's data is determined by the input data x_i of the current time period and the previous hidden state h_{t-1} of LSTM. As shown in the middle part of Fig. 3, the blue module is the e_t^i calculation module and the red part is the softmax function. After the calculation, the attention weight value α_t^i is obtained.

$$\alpha_t^i = \text{Softmax}\left(e_t^i\right) = \frac{\exp\left(e_t^i\right)}{\sum_{k=1}^{T}\exp\left(e_t^i\right)} \tag{9}$$

$$e_t^i = v_d \tanh\left(W_d h_{t-1} + U_d x_i\right) \tag{10}$$

In this context, $v_d \in R^m$, $W_d \in R^{m\times k}$, and $U_d \in R^{m\times m}$ represent the parameters to be trained within the attention module, m denotes the size of both the output of the CNN module and the corresponding input data, and k refers to the size of the LSTM hidden layer.

The attention weights, obtained through normalized softmax probabilities, induce a convex combination of the input vectors. This linear transformation projects the original high-dimensional data into a latent space where salient features are geometrically amplified, as formalized by:

$$x_t' = \sum_{i=1}^{T}\alpha_t^i x_i \tag{11}$$

The feature vector x_t' is subsequently processed by the LSTM, generating an output sequence aligned with the hidden state representation h_t:

$$h_t = f_{\text{LSTM}}\left(h_{t-1}, x_t'\right) \tag{12}$$

The predicted result can be obtained with the hidden state h_T as:

$$\hat{y} = f_{\text{linear}}\left(h_T, x_T'\right) = W_y\left[h_T; x_T'\right] + b_y \tag{13}$$

2.3 CNN-ALSTM

Fig.4 shows the overall model structure.

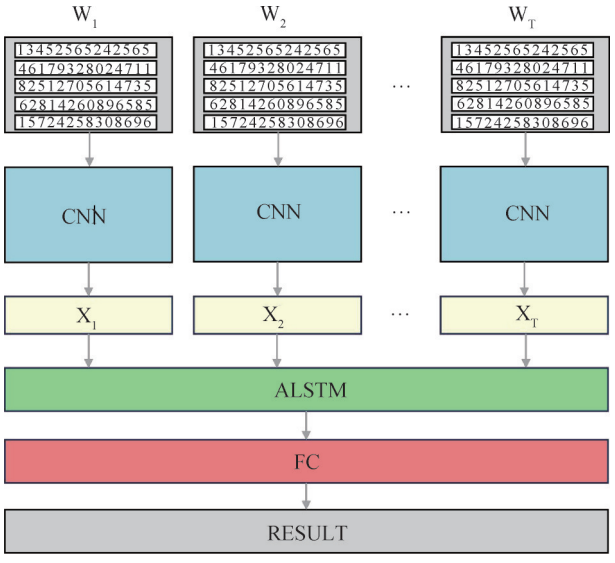

Fig.4 CNN-ALSTM Model Structure

The production data of wafers w_1 to w_T are sequentially input into the CNN network, which generates a one-dimensional feature map x_1 to x_T by extracting dynamic time features and inter-variable

relationships. This feature map is then fed into the ALSTM module as time-series data, where different weights are assigned to data from different time periods. Information integration is performed, and the resistivity gradient prediction of the wafers is then output by the fully connected layer.

The input data, w_t, consists of the recorded growth parameters of a wafer. After formatting, w_t corresponds to the 10×12 matrix of production data for the t-th wafer. The rows represent parameter data recorded chronologically from top to bottom, while each column corresponds to a distinct parameter. The specific parameters are described in the following section.

3 Data Processing

3.1 Data Presentation

The data used in this paper were provided by a Chinese monocrystalline silicon producer. It includes production data from ten ingots of the same specification, produced in the same furnace between June and September 2023, totaling 299, 819 records with 34 parameters each, reflecting growth process characteristics. Eight ingots' data were used for training, and two for testing. The experiment aims to predict the resistivity gradient of silicon wafers, with each wafer having multiple growth data points but one resistivity gradient value.

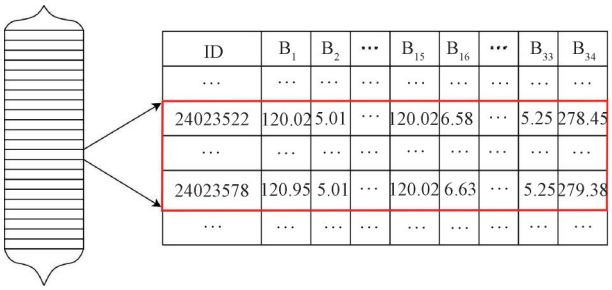

Fig.5 Wafer-specific raw production data

Fig.5 illustrates the relationship between parameters and wafers, where "ID" is the time unit ordinal number and "Bi" represents parameters recorded in the Cz monocrystalline furnace.

During the monocrystalline silicon pulling process, the furnace records a total of 34 parameters. The correspondence between each parameter's code and its name is shown in Table 2.

Table 2 Furnace recorded feature parameters

ID	Feature Name	ID	Feature Name	ID	Feature Name
B_1	Equal diameter total length(mm)	B_{13}	Heater temperature setpoint(℃)	B_{25}	Melt level(mm)
B_2	Sub-furnace pressure(Torr)	B_{14}	Leak rate(mbar·L/s)	B_{26}	Melt level setpoint(mm)
B_3	Crucible stroke(mm)	B_{15}	Crystal length(mm)	B_{27}	Melt temperature(℃)
B_4	Crucible rise(mm)	B_{16}	Crystal weight(kg)	B_{28}	Remaining total weight(kg)
B_5	Crucible rise setpoint(mm)	B_{17}	Crystal rotation(rpm)	B_{29}	Diameter(mm)
B_6	Crucible tracking ratio	B_{18}	Crystal rotation setpoint(rpm)	B_{30}	Diameter setpoint(m)
B_7	Crucible position(mm)	B_{19}	Pull out weight(kg)	B_{31}	Weight(kg)
B_8	Crucible rotation(rpm)	B_{20}	Pulling speed(mm/h)	B_{32}	Weight change rate(kg/h)
B_9	Crucible rotation setpoint(rpm)	B_{21}	Pulling speed setpoint(mm/h)	B_{33}	Main furnace pressure(Torr)
B_{10}	Heater power(KW)	B_{22}	Average growth rate(mm/h)	B_{34}	Seed position(mm)
B_{11}	Heater power setpoint(KW)	B_{23}	Gas flow rate(sccm)		
B_{12}	Heater temperature(℃)	B_{24}	Gas flow rate setpoint(sccm)		

3.2 Data Preprocessing

3.2.1 Feature Selection

The collected Cz monocrystalline furnace recording data may contain redundant information within the feature parameters. To mitigate errors caused by these redundancies and reduce unnecessary subsequent calculations, feature selection on the data is necessary. Initially, the constant data B_6, B_7, B_{14}, B_{23}, B_{24}, B_{25}, B_{26}, and B_{27}, as well as the manually recorded data B_{15}, which is identical to the Cz monocrystalline furnace records, must be deleted. Subsequently, the spearman correlation

coefficient was employed to assess the degree of correlation between the features and to eliminate redundant parameters. Finally, each feature's importance is evaluated using random forest, and unimportant features are removed.

Spearman's correlation coefficient has proven to be an effective method for identifying correlations between parameters. The heat map of correlation coefficients among the feature parameters is presented in Fig.6.

When the correlation coefficient between two parameters exceeds 0.95 or falls below -0.95, it indicates a significant level of redundancy in the information provided by these parameters. Consequently, eliminating one of these parameters becomes necessary. After this removal, the remaining feature parameters are presented in Table 3.

The feature selection principle using Random Forest involves evaluating each feature's contribution to every tree in the Random Forest, calculating the mean importance, and comparing these means to identify relatively unimportant features. After assessing all importance weights, each variable is ranked in descending order of importance, as illustrated in Fig.7.

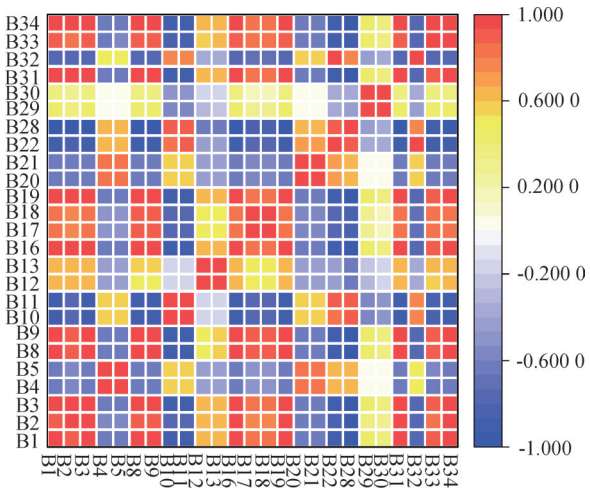

Fig.6 Correlation heatmap of features

Table 3 Selected Feature parameters

ID	Feature name	ID	Feature name	ID	Feature name
B_1	Equal diameter total length(mm)	B_{11}	Heater power setpoint(KW)	B_{22}	Average growth rate(mm/h)
B_2	Sub-furnace pressure(Torr)	B_{13}	Heater temperature setpoint(°C)	B_{28}	Remaining total weight(kg)
B_5	Crucible rise setpoint(mm)	B_{18}	Crystal rotation setpoint(rpm)	B_{30}	Diameter setpoint(m)
B_9	Crucible rotation setpoint(rpm)	B_{21}	Pulling speed setpoint(mm/h)	B_{33}	Main furnace pressure(Torr)

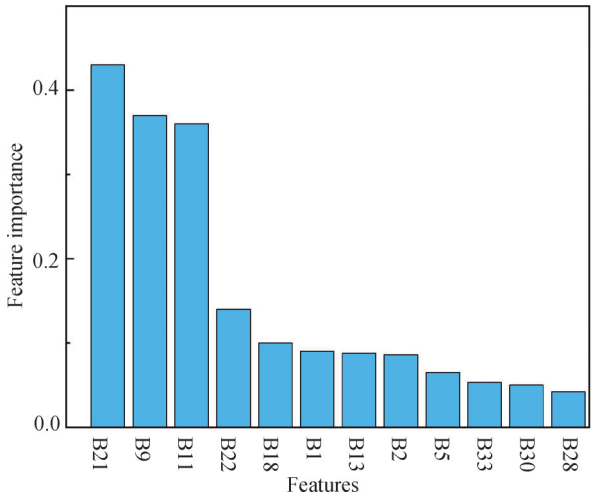

Fig.7 Feature Importance Ranking

In this step, all features contribute to the results, so no further elimination of feature parameters is performed.

3.2.2 Data Normalization

As shown in Fig.5, the data ranges of the parameters collected by different sensors vary significantly. For instance, parameter B_2 ranges from 5 to 7, while parameter B_{34} ranges from 250 to 300. This disparity in magnitudes can negatively impact model training, as larger parameter values may disproportionately influence the model, even if they are less important. Additionally, excessive variation between features can cause large fluctuations in the loss function during training, hindering convergence to an optimal solution. To ensure balanced contributions from each feature and improve both training and prediction performance, normalization of the training data is necessary. In this paper, we apply min-max normalization, mapping each feature to the range of 0 to 1 as shown in Equation (14).

$$x_m = \frac{x_o - min}{\max - min} \tag{14}$$

In the formula, x_m denotes the normalized data, x_o denotes the original data, *min* is the minimum value of the original data, and *max* is the maximum value of the original data.

3.2.3 Data Reshaping

Since the entry point of the prediction model in this paper is a CNN, the production data for each wafer must be formatted into the same shape. Due to the high density of the collected production data, adjacent data points may be identical. Therefore, duplicate data are removed, and the data for each wafer are transformed into two-dimensional arrays with an equal number of rows. As all wafer production data come from the same Cz monocrystalline furnace and undergoes the same preprocessing steps, the data for each wafer have the same number of columns and do not require further processing.

In this paper, we eliminate the data with the highest

degree of repetition. Adjacent data that has more unchanged parameters is selected to delete several of these adjacent data. To achieve this, we first calculate the degree of repetition for each data point relative to others. The greater the similarity between two data points, the higher the degree of repetition. This results in a data repetition matrix, as shown in Fig.8. The value in the *i*-th row and *j*-th column of the matrix represents the degree of repetition between the *i*-th and *j*-th data points in the silicon wafer production data.

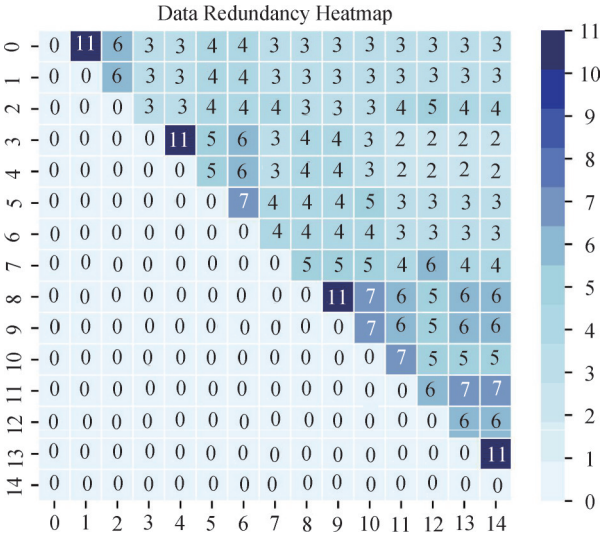

Fig.8 Data Redundancy Heatmap

Since the minimum number of rows in the collected two-dimensional data is ten, data points with high repetition are removed, leaving only ten rows. The production data for each wafer is then organized into a matrix w of size [10×12].

3.2.4 Time Series Data Construction

To adapt the LSTM module in the CNN-ALSTM model, the input data must be formatted as time series. Each time step corresponds to the production data matrix of a wafer, which is transformed into one-dimensional data after feature extraction by the CNN module. A sliding window is then applied, where the window region serves as the input time series data and the corresponding resistivity gradient value. The window is moved one wafer position at a time, generating new samples until it reaches the final wafer position. This process produces the model's input data set and target value data set. The construction process is illustrated in Fig.9.

4 Results and Discussion

4.1 Sliding Window Selection

The sliding window lengths were selected through empirical trials with values set at 5, 10, 15, 20, 25, and 30 for testing. After 150 training rounds, the testing loss values for each window length were compared to identify the most efficient length. The relationship between testing loss values and window lengths is shown in Fig.10.

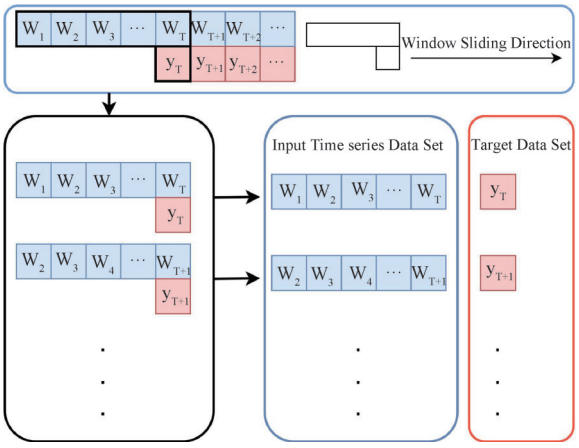

Fig.9 Construction of Time Series Data Chart

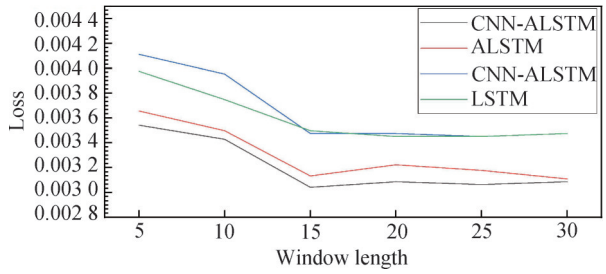

Fig.10 Relationship Between Testing Loss and Window Length

As shown in the figure, when the window length reaches 20, the loss value does not significantly decrease, and in some models, it even increases. This is likely due to the introduction of redundant information, which reduces the efficiency of stochastic gradient descent, leading to a higher final loss. Therefore, a window length of 15 was selected, using the production data from the previous 14 wafers, along with the production data of the 15th wafer, to predict its resistivity gradient value.

4.2 Ablation Experiment

To evaluate the contribution of each module in the model, ablation experiments were conducted to assess the impact of removing individual modules on model performance. Four experimental setups with different model structures were designed, as shown in Table 4. The parameters for the same modules were kept consistent across all model variants. The CNN parameters followed the previous description, and the LSTM was configured as a single-layer network with a hidden layer size of 64, and the models were trained for 150 epochs with a learning rate of 0.001. The prediction accuracy, convergence speed, and training time of the CNN-ALSTM, ALSTM, CNN-LSTM, and LSTM models were compared. The models were implemented using PyTorch, with MSE as the loss function and Adam as the optimizer.

The change of loss values for each testing dataset during training is shown in Fig.11.

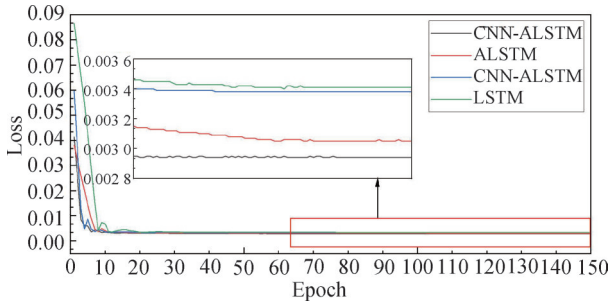

Fig.11 Test Set Loss Curves for Each Model

The loss value curves in the test set show that the final loss values for both ALSTM and CNN-ALSTM models are much smaller than those for LSTM and CNN-LSTM models. This confirms the effectiveness of the attention module in improving prediction accuracy for this task. The final loss values for both CNN-LSTM and CNN-ALSTM models are slightly smaller than those for LSTM and ALSTM models, and the loss curves for CNN-LSTM and CNN-ALSTM show greater curvature in the first ten epochs compared to LSTM and ALSTM. This finding suggests that CNN is able to improve the prediction accuracy of the models and speed up the overall convergence of the models.

The final loss values and the iteration time per epoch for each model are recorded in Table 4.

Table 4 Experimental Group Structures and Results

Group No.	CNN	Attention	MSE	Training time (s/epoch)
1	√	√	0.0030	70.45
2	-	√	0.0031	27360.84
3	√	-	0.0034	0.81
4	-	-	0.0035	3.07

The MSE of the first group was 0.0030, 11.76% lower than the 0.0034 of the third group, and the MSE of the second group was 0.0031, 11.43% lower than the 0.0035 of the fourth group. This comparison highlights the critical role of the Attention module in improving model prediction accuracy, validating its effectiveness. The Attention module captures the time lag, a key variable in the production process, and transmits this information to the LSTM, enhancing its ability to predict time-series data. Meanwhile, the first group's MSE was 3.23% lower than that of the second group, and the third group's MSE was 2.86% lower than that of the fourth group. That means the CNN module can also improve the predictive accuracy of models. This is attributed to its capability to extract short-term dynamic temporal features and spatial characteristics from production data, as well as to eliminate high-frequency noise. The iteration time of the first group was 70.45 seconds, 87 times longer than that of the third group, and the second group's iteration time was 27360.84 seconds, 8912 times longer than the fourth group's. This suggests that the attention mechanism dominates the computational overhead, accounting for over 85% of the total inference latency. However, the iteration time of the first group was reduced by 387 times compared to the second group, and that of the third group by 2.8 times compared to the fourth group. These results demonstrate that the CNN module not only enhances model prediction accuracy but also enables the effective extraction of discriminative features from high-dimensional production data. By filtering redundant information, the CNN significantly reduces the dimensionality of the input data processed by subsequent attention mechanisms. This reduction not only alleviates computational burden but also accelerates both training convergence and inference speed.

4.3 Comparative Experiment

To verify the effectiveness of the proposed model, a comparative experiment was conducted. Three widely used regression models—RNN, SVR, XGBoost, FNN, and GRU—were selected for comparison. The RNN and GRU models used the same hyperparameters as the proposed model, while the SVR, XGBoost, and FNN models employed grid search optimization to identify the optimal parameters.

Fig. 12 shows a comparison of the partial prediction results from each model with the actual results. As observed, the predictions from the CNN-ALSTM model are closer to the original data and exhibit greater stability compared to those from the other models. As shown in Table 5, the CNN-ALSTM model achieves the lowest mean squared error (MSE) of 0.0030, indicating the smallest average squared difference between predicted and actual values. It also achieves the lowest mean absolute error (MAE) of 0.0431, reflecting its ability to capture the true variability of the data with greater precision and minimize prediction error. Additionally, the model has an R^2 value of 0.9071, demonstrating its capacity to explain the most variability in the data.

To further highlight the superiority of the CNN-ALSTM model, the prediction errors are visualized to facilitate a comparative analysis of the models' prediction performance.

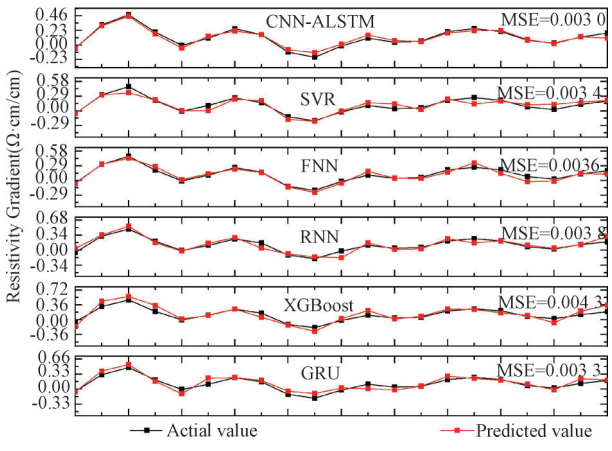

Fig.12 Charts Comparing Predicted Values to Actual Values

Table 5 Metric Values for Each Model

	CNN-ALSTM	SVR	FNN	RNN	XGBoost	GRU
MSE	0.0030	0.0034	0.0036	0.0038	0.0043	0.0033
MAE	0.0431	0.0474	0.0472	0.0493	0.0521	0.0473
R^2	0.9071	0.8948	0.8887	0.8817	0.8675	0.8966

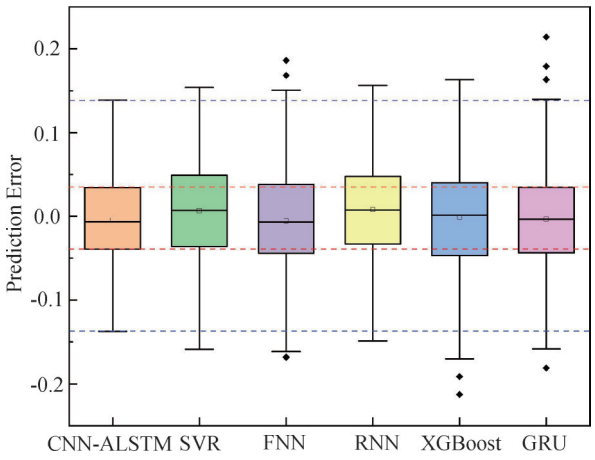

Fig.13 Box Plot Comparing Prediction Errors

Fig.13 illustrates the prediction error distribution for each model in the form of a box plot. The CNN-ALSTM model shows a more concentrated distribution around the median, indicating the CNN-ALSTM model demonstrates lower prediction errors and more consistent performance, with predictions that are less prone to large deviations. This may be attributed to the Attention mechanism, which effectively captures changes in the production data, allowing the model to respond more accurately to feature variations. In contrast, the RNN model displays a broader error distribution, suggesting higher prediction errors and greater variability in its performance. This difference can be attributed to the case study's significant time lag, which the RNN, with its limited ability to retain long-term information, may struggle to capture. This likely led to the loss of crucial information, resulting in the observed performance disparity. These results validate the superior performance of the CNN-ALSTM model.

5 Conclusion

The main contributions of this paper are summarized as follows:

(1)This paper proposes a CNN-ALSTM hybrid soft-sensing model to address the critical gap in real-time resistivity gradient monitoring for industrial monocrystalline silicon production. The soft-sensing model achieves high accuracy in predicting the resistivity gradient of monocrystalline silicon in the pulling process of monocrystalline silicon, with an average prediction error of just 0.0030. This suggests significant potential for monitoring and optimizing process parameters and improving resistivity uniformity.

(2)This study proposes a CNN-ALSTM hybrid model to address the dynamic time-lag challenge in monocrystalline silicon production processes. The model incorporates an adaptive time-lag compensation mechanism that enables dynamic parameter adjustment and optimization. Experimental validation demonstrates significant enhancements in resistivity gradient prediction accuracy, achieving performance improvements of 9.09% -30.23% compared to conventional approaches (RNN, GRU, etc.) while maintaining real-time monitoring capabilities. This advancement effectively mitigates measurement deviations caused by dynamic time-lag during industrial production.

Building upon these contributions, several promising research directions emerge for further development. Future research directions may include real-time optimization of resistivity gradients through dynamic adjustment of production process parameters using the proposed CNN-ALSTM framework.

Author Contribution:

Zhiheng Zhang: Took the lead in developing the soft-sensing model and overall study design. Conducted extensive raw data collection and led data analysis efforts. The core arguments and main findings of the paper are presented. Wrote most of the manuscript. Zengguo Tian: Guide the research direction and provide valuable insights at every stage. Conducted Revision of the manuscript.

Funding Information:

This research received no external funding.

Data Availability

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request.

Conflicts of Interest:

The authors declare no competing interests.

Dates:

Received 04 March 2025; Accepted 18 May 2025; Published online 30 June 2025

References

[1] Ren J. -C., D. Liu, and Y. Wan. (**2021**). Modeling and application of Czochralski silicon single crystal growth process using hybrid model of data-driven and mechanism-based methodologiesReference [J]. *Journal of Process Control* . 104, 74-85. doi:10.1016/j.jprocont.2021.06.002

[2] Liu D., X.-G. Zhao, and Y. Zhao. (**2017**). A review of growth process modeling and control of Czochralski silicon single crystalReference [J]. *Control Theory and Applications* . 34(1),

1-12. doi:10.7641/CTA.2017.60247

[3] Guo R., et al. (**2021**). Weld Defect Detection From Imbalanced Radiographic Images Based on Contrast Enhancement Conditional Generative Adversarial Network and Transfer LearningReference [J]. *IEEE Sensors Journal* . 21(9), 10844-10853. doi:10.1109/JSEN.2021.3059860

[4] Chhantyal K., et al. (**2018**). Upstream Ultrasonic Level Based Soft Sensing of Volumetric Flow of Non-Newtonian Fluids in Open Venturi ChannelsReference [J]. *IEEE Sensors Journal* . 18(12), 5002-5013. doi:10.1109/JSEN.2018.2831445

[5] Yuan X., et al. (**2018**). Multi-similarity measurement driven ensemble just-in-time learning for soft sensing of industrial processesReference [J]. *Journal of Chemometrics* . 32(9), e3040. doi:10.1002/cem.3040

[6] Yin S., L. Liu, and J. Hou. (**2016**). A multivariate statistical combination forecasting method for product quality evaluationReference [J]. *Information Sciences* . 355-356, 229-236. doi:10.1016/j.ins.2016.03.035

[7] Ge Z., et al. (**2014**). Two-level PLS model for quality prediction of multiphase batch processesReference [J]. *Chemometrics and Intelligent Laboratory Systems* . 130, 29-36. doi:10.1016/j.chemolab.2013.09.008

[8] Guo R., et al. (**2024**). Adversarial Robustness Enhancement for Deep Learning-Based Soft Sensors: An Adversarial Training Strategy Using Historical Gradients and Domain AdaptationReference [J]. *Sensors* . 24(12), 3909. doi:10.3390/s24123909

[9] Guo R., et al. (**2023**). A Self-Interpretable Soft Sensor Based on Deep Learning and Multiple Attention Mechanism: From Data Selection to Sensor ModelingReference [J]. *IEEE Transactions on Industrial Informatics* . 19(5), 6859-6871. doi:10.1109/TII.2022.3181692

[10] Liu X., et al. (**2020**). Transient global modeling for the pulling process of Czochralski silicon crystal growth. II. Investigation on segregation of oxygen and carbonReference [J]. *Journal of Crystal Growth* . 532, 125404. doi:10.1016/j.jcrysgro.2019.125404

[11] Liu X., et al. (**2020**). Transient global modeling for the pulling process of Czochralski silicon crystal growth. I. Principles, formulation, and implementation of the modelReference [J]. *Journal of Crystal Growth* . 532, 125405. doi:10.1016/j.jcrysgro.2019.125405

[12] Kutsukake K., et al. (**2020**). Real-time prediction of interstitial oxygen concentration in Czochralski silicon using machine learningReference [J]. *Applied Physics Express* . 13 (12), 125502. doi:10.35848/1882-0786/abc6ec

[13] Qi X., et al. (**2020**). Optimization of the melt/crystal interface shape and oxygen concentration during the Czochralski silicon crystal growth process using an artificial neural network and a genetic algorithmReference [J]. *Journal of Crystal Growth*. 548, 125828. doi:10.1016/j.jcrysgro.2020.125828

[14] Zhang J., et al. (**2020**). A deep learning based dislocation detection method for cylindrical crystal growth processReference [J]. *Applied Sciences* . 10(21), 7799. doi:10.3390/app10217799

[15] Yuting S. and L. Hongxing. (**2023**). A deep learning based dislocation detection method for cylindrical silicon growth processReference [J]. *Applied Intelligence* . 53(8), 9188-9203. doi:10.1007/s10489-022-03800-0

[16] Jiang L., R. Xue, and D. Liu. (**2023**). Node-Loss Detection Methods for CZ Silicon Single Crystal Based on Multimodal Data FusionReference [J]. *Sensors* . 23(13), 5855. doi:10.3390/s23135855

[17] Liu D., et al. (**2019**). Nonlinear generalized predictive control of the crystal diameter in CZ-Si crystal growth process based on stacked sparse autoencoderReference [J]. *IEEE Transactions on Control Systems Technology* . 28(3), 1132-1139. doi:10.1109/TCST.2019.2898975

[18] Feng Z., et al. (**2023**). Bi-LSTM-Based Dynamic Prediction Model for Pulling Speed of Czochralski Single-Crystal FurnaceReference [J]. *Journal of Computing and Information Science in Engineering* . 23(4), 041010. doi:10.1115/1.4056138

[19] Huang W., G. Zhang, and J. Wang. (**2022**). Multiobjective Optimization of Process Parameters of Silicon Single Crystal GrowthReference [J]. *Wireless Communications and Mobile Computing* . 2022, 1-10. doi:10.1155/2022/5312590

[20] Wan Y., et al. (**2021**). Data-driven model predictive control of Cz silicon single crystal growth process with V/G value soft measurement modelReference [J]. *IEEE Transactions on Semiconductor Manufacturing* . 34(3), 420-428. doi:10.1109/TSM.2021.3088855

[21] Ren J.-C., D. Liu, and Y. Wan. (**2021**). Model-free adaptive iterative learning control method for the czochralski silicon monocrystalline batch processReference [J]. *IEEE Transactions on Semiconductor Manufacturing* . 34(3), 398-407. doi:10.1109/TSM.2021.3074625

[22] Dang Y., L. Liu, and Z. Li. (**2019**). Optimization of the controlling recipe in quasi-single crystalline silicon growth using artificial neural network and genetic algorithmReference [J]. *Journal of Crystal Growth* . 522, 195-203. doi:10.1016/j.jcrysgro.2019.06.033

[23] Boucetta A., et al. (**2019**). Application of artificial neural network to optimize sensor positions for accurate monitoring: An example with thermocouples in a crystal growth furnaceReference [J]. *Applied Physics Express* . 12(12), 125503. doi:10.7567/1882-0786/ab52a9

[24] Livieris I.E., E. Pintelas, and P. Pintelas. (**2020**). A CNN – LSTM model for gold price time-series forecastingReference [J]. *Neural computing and applications* . 32, 17351-17360. doi:10.1007/s00521-020-04867-x

[25] Krizhevsky A., I. Sutskever, and G. E. Hinton. (**2017**). Imagenet classification with deep convolutional neural networksReference [J]. *Communications of the ACM* . 60(6), 84-90. doi:10.1145/3065386

[26] Kim, T.-Y. S.-B. Cho. (2019). Predicting residential energy consumption using CNN-LSTM neural networksReference [J]. *Energy* . **182**, 72-81. doi:10.1016/j.energy.2019.05.230

[27] Niu Z., G. Zhong, and H. Yu. (**2021**). A review on the attention mechanism of deep learningReference [J]. *Neurocomputing*. 452, 48-62. doi:10.1016/j.neucom.2021.03.091

[28] Dong Z., et al. A Multiphase Dual Attention-Based LSTM Neural Network for Industrial Product Quality PredictionReference [J]. *IEEE Transactions on Industrial Informatics* . 20(7), 9298-9307. doi:10.1109/TII.2024.3383054