RESEARCH ARTICLE

# Enhanced Air Pollution Prediction via Adam-Optimized Multi-Head Attention and Hybrid Deep Learning

**Chenbin Gu**[1], **Yimi Tan**[2], **Xiaoqi Yin**[1,*], **Xuejun Li**[1], **Yudong Yang**[1] **and Yan Lv**[3]

[1] School of Electronic Information Engineering, Huaiyin Institute of Technology, Huaian 223003, China
[2] School of Atmospheric Sciences, Nanjing University of Information Science and Technology, Nanjing 210044, China
[3] School of Remote Sensing and Surveying Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China

## Abstract

**To address the challenge of traditional models in simultaneously capturing local fluctuations and global trends for air pollutant concentration prediction, this paper proposes a multimodal deep learning model named MLP-BiLSTM- MHAT. The model integrates static features via MLP, extracts temporal dependencies through bidirectional LSTM (BiLSTM), and employs a Multi-head Attention mechanism (MHAT) to fuse local and global features while enhancing interactions between static and temporal characteristics. An improved Adam algorithm dynamically optimizes learning rates to balance the influence of heterogenous features. Validated on multi-site air quality data from Beijing, experimental results demonstrate that MLP-BiLSTM-MHAT outperforms baseline models with an average reduction of 1.9% in RMSE, 4.2% in MAE, and a 1.8% improvement in R², showcasing superior accuracy and robustness across diverse pollutants and scenarios.**

**Keywords**: multimodal, deep learning network, improved Adam algorithm, air pollutant concentration prediction.

## 1 Introduction

Air pollution is an important public health issue worldwide [1], and air pollutants have a significant impact on human health, especially the respiratory system [2, 3]. This makes high-precision air quality prediction an urgent need for public health protection and environmental management decisions. Currently, there are three main methods for predicting the concentration of air pollutants: numerical forecasting [4], statistical methods, and machine learning [5]. Numerical forecasting uses observed data to establish atmospheric chemical and physical models, simulating the transmission, diffusion, reflection, and deposition processes of pollutants in the atmosphere [6]. This method is based on atmospheric dynamics equations and pollutant chemical reaction mechanisms [7], and can simulate the three-dimensional spatiotemporal distribution characteristics of pollutants at the regional scale. Statistical methods are represented by autoregressive

**Academic Editor:**
Jianlei Kong

**Submitted:** 25 September 2025
**Accepted:** 23 October 2025
**Published:** 29 January 2026

**Vol.** 3, **No.** 1, 2026.
10.62762/TIS.2025.951370

***Corresponding author:**
✉ Xiaoqi Yin
hy_xuebao2009@126.com

**Citation**
Gu, C., Tan, Y., Yin, X., Li, X., Yang, Y., & Lv, Y. (2026). Enhanced Air Pollution Prediction via Adam-Optimized Multi-Head Attention and Hybrid Deep Learning. *ICCK Transactions on Intelligent Systematics*, 3(1), 11–20.

moving average models (ARIMA) [8] and generalized additive models (GAM) [9]. Gourav et al. [10] treated historical changes in air pollutant concentrations as a time series, modeled them using ARIMA, and predicted the air quality of New Delhi, India for future months and quarters. Cortina–Januchs et al. [11] used clustering algorithms to find the relationship between PM10 and meteorological variables, and then used multi-layer regression to predict the concentration of PM10. They found that meteorological variables are an important factor in air pollution prediction, but ignored nonlinear relationships, resulting in low accuracy of the model prediction.

Compared with numerical forecasting and statistical methods, machine learning methods, especially deep learning methods, are more efficient in the field of pollutant prediction. They can generate more complex models to support dynamic and unstable data, achieving accurate multi-scale air quality prediction. The Universal Approximation Theorem proposed by Hornik et al. [12] provides a core theoretical basis for the application of MLP in static feature extraction. This theory ensures that MLP has strong nonlinear mapping capabilities, sufficient to learn any complex potential relationship between static features and target pollutant concentrations. The bidirectional long short-term memory network (BiLSTM) proposed by Graves et al. [13] breaks through the inherent limitations of traditional unidirectional recurrent neural networks in processing temporal information by introducing bidirectional recurrent neural networks. Compared to unidirectional LSTM, this bidirectional structure can more comprehensively capture the dynamic evolution patterns of time-series data such as air pollutant concentrations. Yang et al. [14] used MLP for CO2 emission prediction, achieving accurate prediction of road $CO_2$ emissions with high spatiotemporal resolution. Aamir et al. [15] applied BiLSTM to the field of carbon emission prediction, predicting environmental changes in South Asian carbon emission patterns and analyzing emission trends and influencing factors in China and South Asian countries. The Transformer model proposed by Vaswani et al. [16] pioneered the use of self attention mechanism in sequence modeling, and its core multi head attention mechanism [17] can capture diverse dependency relationships in parallel. Dai et al. [18] introduced Transformer into the field of air quality prediction and verified the advantages of attention mechanism in long-range dependency modeling. Yu et al. [19] proposed the Temporal Convolutional Network (TCN), which effectively solves the dilemma of traditional sequence models in long-range dependency modeling by integrating causal convolution and dilated convolution, and introducing residual connections. Li et al. [20] proposed the TCN-BiLSTM-DMAttention model and applied it to predict air pollutants, achieving 1-hour prediction of future air pollutants.

It is worth noting that advanced architectures represented by Transformer and TCN have made breakthroughs in long-range dependency modeling and computational efficiency, but their original design intention is mostly to handle homogeneous temporal data. In practical air quality prediction services, data is essentially multimodal, containing both static features and pollution time series features. Most existing research adopts the method of feature flattening concatenation, which forcibly concatenates different features into the same vector space, but fails to achieve deep level interaction between different features, resulting in insufficient ability of the model to capture the main factors when dealing with the sudden increase of pollution caused by static factors and the accumulation process of temporal factors. This problem limits the further improvement of prediction accuracy. To overcome the above problems, this study innovatively proposes a multimodal deep learning [21] model MLP-BiLSTM-MHAT, which achieves deep level interaction and fusion of static features and temporal features. By introducing storage units to improve the Adam algorithm, the problem of multimodal gradient conflicts has been effectively alleviated. Through comparative experiments with traditional recursive networks such as GRU and LSTM, as well as advanced architectures such as TCN and Transformer, the effectiveness of this model in predicting various air pollutants has been verified.

## 2 Model Introduction

### 2.1 MLP

MLP(Multi-layer Perceptron), Multi-layer Perceptron is a common supervised learning neural network model. The following Figure 1 shows the network structure of MLP. It can be seen from the figure that MLP adopts a hierarchical structure of layered stacking, including input layer, hidden layer, and output layer. The data propagates layer by layer from the input layer and eventually reaches the output layer.

In this study, MLP was used to capture the complex interactions between pollutants and meteorological

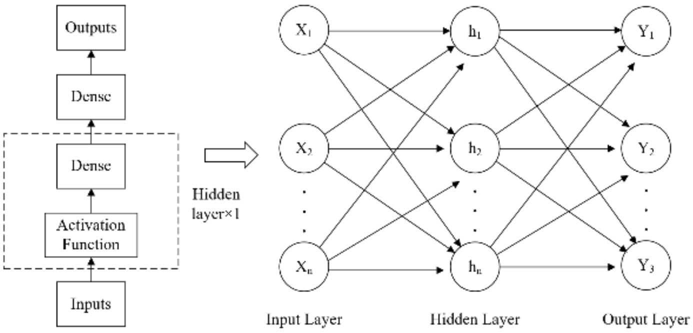


**Figure 1.** The network structure of MLP.

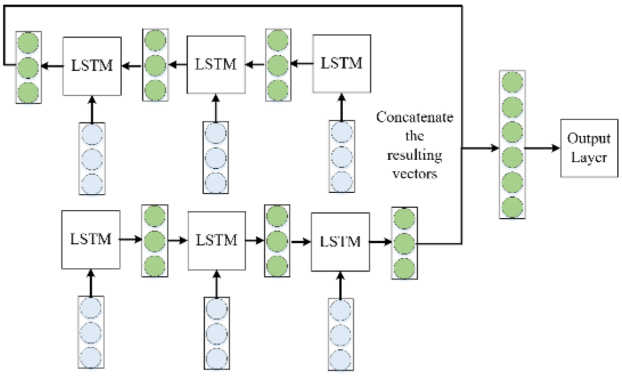


**Figure 2.** The network structure of BiLSTM.

factors and extract static features. The output calculation formula of the MLP layer is as follows:

$$h^{(n)} = f(W^{(n)}h^{(n-1)} + b^{(n)}) \tag{1}$$

In the formula, $h^{(n)}$ represents the output feature vector of the $n$-th layer of MLP; $W^{(n)}$ represents the weight matrix of the $n$-th layer; $b^{(n)}$ represents the bias term of the $n$-th layer; $f$ represents the activation function.

After MLP layer processing, output static feature vectors:

$$H_{\text{static}} \in \mathbb{R}^{d_{\text{mlp}}} \tag{2}$$

In the formula, $d_{\text{mlp}}$ represents the dimension of MLP output features.

### 2.2 BiLSTM

BiLSTM (Bidirectional Long Short Term Memory) is an improved structure based on traditional LSTM, aimed at capturing the contextual dependencies of sequence data more comprehensively. Its core idea is to simultaneously utilize the past and future information of the sequence. As shown in Figure 2, the bidirectional long short-term memory network consists of two independent LSTM layers: one forward LSTM layer processes sequence data in chronological order, and the other reverse LSTM layer processes it in reverse chronological order. These two directions of LSTM will output a hidden state at each time step, and finally merge the forward hidden state and reverse hidden state through concatenation, weighting, and other methods as the final output of that time step.

BiLSTM simultaneously processes modeling forward and backward temporal dependencies through dual channels, with the specific expression being:

$$h_t^{\rightarrow} = \text{LSTM}_z(x_t, h_t^{\rightarrow -1}) \tag{3}$$

$$h_t^{\leftarrow} = \text{LSTM}_f(x_t, h_t^{\leftarrow +1}) \tag{4}$$

In the equation, $h_t^{\rightarrow}$ represents the hidden state of the forward LSTM unit at time step $t$; $h_t^{\leftarrow}$ represents the hidden state of the reverse LSTM unit at time step $t$.

By concatenating the forward and backward hidden states, the feature vectors for each time step are obtained:

$$h_t = [h_t^{\rightarrow}; h_t^{\leftarrow}] \in \mathbb{R}^{2d_{\text{lstm}}} \tag{5}$$

The feature matrix of the final output time series is:

$$H_{\text{seq}} = [h_1, \ldots, h_T]^T \in \mathbb{R}^{T \times 2d_{\text{lstm}}} \tag{6}$$

### 2.3 Multi-head Attention

Multi-head Attention is one of the core components of Transformer models, which enhances the expressive power of the model by capturing diverse dependencies in the input sequence in parallel. As shown in Figure 3, the typical Multi-Head Attention mechanism structure mainly includes three parts: Query, Key, and Value, represented as $Q$, $K$, and $V$ respectively. In practical calculations, the model calculates the similarity between query $Q$ and key $K$ (usually using dot product or scaled dot product), obtains attention weights, and then applies these weights to the value $V$ to generate a weighted sum as the final output.

In this study, the Multi-Head Attention mechanism interacts and fuses static and temporal features. By calculating the attention score through dot product and normalizing it to its weight using Softmax function, the features are finally aggregated based on the weight matrix to achieve feature optimization and enhancement. The specific expression is:

$$A = \text{softmax}\left(\frac{QK^T}{\sqrt{d_a}}\right) \tag{7}$$

$$H' = AV \tag{8}$$

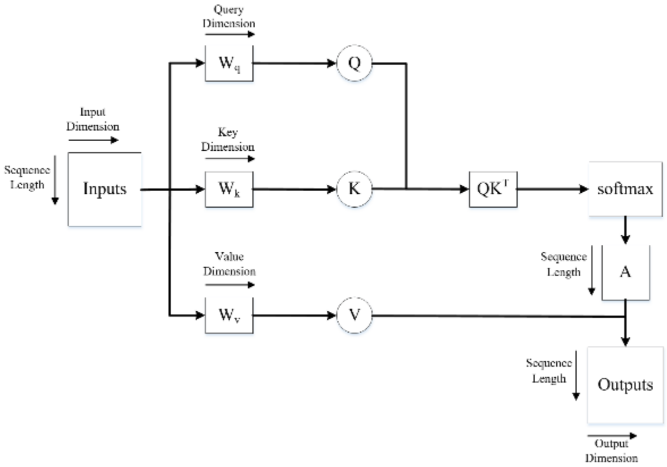


**Figure 3.** The network structure of Multi-head Attention.

Among them, $A$ represents the similarity between each time step; $Q$ represents the query matrix; $K$ represents the key matrix; $V$ represents the value matrix; $H'$ is a weighted representation of each time step.

### 2.4 Adam Algorithm

The Adam (Adaptive Moment Estimation) algorithm [22] is a widely used optimization algorithm in deep learning, which adjusts the learning rate of parameters by calculating the first-order and second-order matrix estimates of gradients. The core calculation steps of the Adam algorithm are as follows:

Step 1: Calculate the gradient, for each parameter, calculate its gradient $g_t$.

Step 2: Update the momentum $m_t$ and second-order matrix $v_t$.

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \tag{9}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \tag{10}$$

In the equation, $\beta_1$ represents the decay rate that controls momentum; $\beta_2$ represents the decay rate that controls the square gradient.

Step 3: Correct the deviation between the first-order matrix and the second-order matrix using the following formula:

$$\hat{m}_t = \frac{m_t}{1-\beta_1} \tag{11}$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2} \tag{12}$$

In the equation, $\hat{m}_t$ is the modified first-order matrix estimate, and $\hat{v}_t$ is the modified second-order matrix estimate.

Step 4: Use the modified parameters to update the gradient, with the specific formula being:

$$\theta_t = \theta_{t-1} - \frac{\partial \hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \tag{13}$$

In the formula, $\theta_t$ represents the updated parameter value, $\alpha$ represents the learning rate, and $\epsilon$ is a small constant.

### 2.5 MLP-BiLSTM-MHAT architecture

The architecture of the MLP-BiLSTM-MHAT model is shown in Figure 4. The model's input layer consists of the air quality data selected for this study. The division of the input data is detailed in Table 1. Initially, static features are extracted through the MLP layer. These features are then processed by the BiLSTM layer, which learns the temporal dependencies in both the forward and backward directions. The multi-head attention layer combines the static features, extracted by the MLP, with the temporal features output by the BiLSTM. Finally, the fused and enhanced features are passed through a fully connected network to calculate the predicted values, which are then output to the output layer.

**Table 1.** Input data division.

| Data type | Input method | Processing layer | Dimension |
|---|---|---|---|
| Time series feature | 3D tensor | BiLSTM | (32,24,256) |
| Static feature | 2D matrix | MLP | (32,256) |

The pseudocode of the MLP-BiLSTM-MHAT model is shown in Algorithm 1.

```
Algorithm 1: MLP-BiLSTM-MHAT
Input: $X = \{X_1, X_2, \ldots, X_a\}$
Output: $Y = \{Y_1, Y_2, \ldots, Y_t\}$
Initialize static feature sequence $S = [\,]$;
for $a = 1$ to $a$ do
    $H_t = \mathrm{MLP}(X_i)$;
$H_{\mathrm{seq}} = \mathrm{BiLSTM}(X_t)$;
$P = \text{Multi-head Attention}(H_t, H_{\mathrm{seq}})$;
$Y = \mathrm{DENSE}(P)$;
return $Y$
```

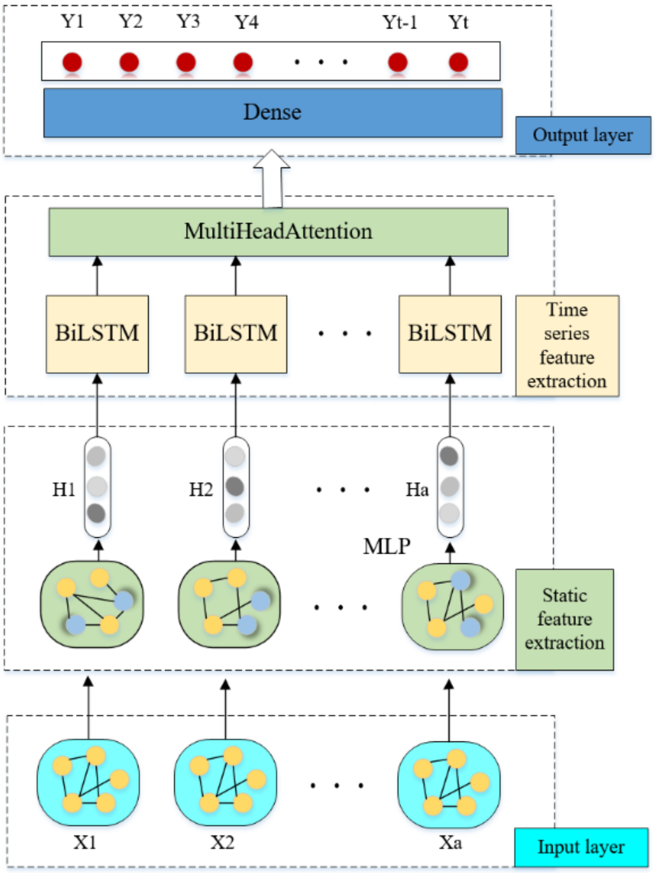


**Figure 4.** Overall framework of MLP-BiLSTM-MHAT model.

## 3 Experiment and Result Analysis

### 3.1 Dataset Source and Preprocessing

This study selected air quality data from Beijing as experimental samples. This dataset contains complete meteorological data and air pollutant data, including 420768 meteorological data and air pollutant data from March 1, 2013 to February 28, 2017, recorded hourly. The data of each site includes meteorological data such as temperature, air pressure, precipitation, and air pollutant data such as PM2.5, PM10, $SO_2$. The detailed data types are shown in Table 2.

**Table 2.** Description of dataset types.

| Data type | Variable Name | Unit |
|---|---|---|
| Meteorological data | temperature | ℃ |
| | atmospheric pressure | hPa |
| | dew point temperature | ℃ |
| | precipitation | mm |
| | wind direction | - |
| | wind speed | m/s |
| Air pollutant data | PM2.5 | $\mu g/m^3$ |
| | PM10 | $\mu g/m^3$ |
| | $SO_2$ | $\mu g/m^3$ |
| | $NO_2$ | $\mu g/m^3$ |
| | CO | $mg/m^3$ |
| | $O_3$ | $\mu g/m^3$ |

For missing values in the original dataset, mean imputation method is used to handle continuous missing values, backward imputation method is used to handle discontinuous missing values, and quartile range method is used to handle outliers in the original dataset. Normalize the missing and outlier processed datasets, and finally divide the dataset into training, validation, and testing sets in a ratio of 7:2:1.

### 3.2 Improving Adam algorithm to dynamically optimize learning rate

In this study, the Adam algorithm was improved by introducing memory units and used to dynamically optimize the learning rate. Compared with the core computational steps of the Adam algorithm in Section 2.4, the specific improvements are as follows:

Step 1: Calculate the gradient and introduce memory units. For each parameter, calculate its gradient $g_t$, and each $\theta_t$ has a corresponding memory unit $M_t$.

Step 2: Use the forgetting factor and update factor to update the memory unit, with the formula:

$$M_t = f_t M_{t-1} + u_t g_t \tag{7}$$

$$f_t = \frac{1}{1 + \exp(-a(g_t - \mu))} \tag{8}$$

$$u_t = \frac{1}{1 + \exp(-b(g_t - \mu))} \tag{9}$$

where $g_t$ is the gradient of the current time step $t$; $M_t$ is the memory state of the current time step; $f_t$ is the forgetting factor; $u_t$ is the update factor; $a$ and $b$ are hyperparameters; $\mu$ is the reference value.

Step 3: Update the momentum $m_t$ and second-order moment $v_t$, add memory information, and the formula is:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t + \gamma M_t \tag{17}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{18}$$

where $\gamma$ is a hyperparameter.

Step 4: Correct the deviation between the first-order matrix and the second-order matrix using the following formula:

$$\hat{m}_t = \frac{m_t}{1-\beta_1} \tag{19}$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2} \tag{20}$$

Step 5: Use the corrected parameters to update the gradient, with the specific formula being:

$$\theta_t = \theta_{t-1} - \frac{\partial \hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} = \theta_{t-1} - \frac{\partial}{\sqrt{\hat{v}_t + \epsilon}} \tag{21}$$

This study compared the performance of the original Adam and the improved Adam during the training process of the MLP-BiLSTM-MHAT model to verify the optimization effect of the improved Adam algorithm on the model training process. As shown in Figures 5 and 6, it is evident from the validation loss curve that the improved algorithm outperforms the original Adam in terms of convergence speed and training stability.

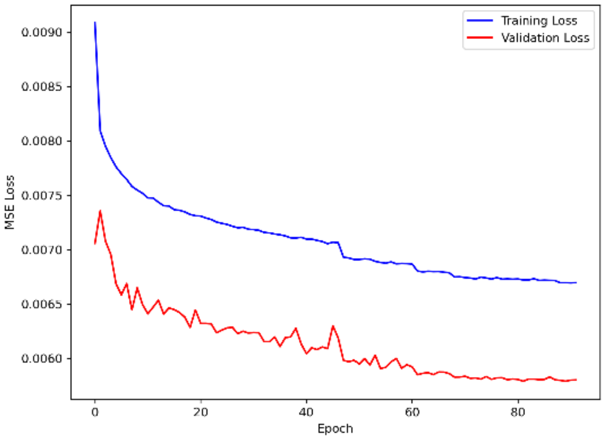


**Figure 5.** Training loss and validation loss curves of the model before improvement.

Table 3 compares the quantitative performance of the MLP-BiLSTM-MHAT model before and after the improvement of the Adam algorithm. Among them, O-Adam stands for Original Adam; I-Adam stands for Improved Adam; IA stands for Improvement Amplitude; CIT stands for Convergence iteration times; OVL stands for Optimal Verification Loss; SD stands for Standard Deviation. The Standard Deviation is calculated based on the last 25 training epochs.

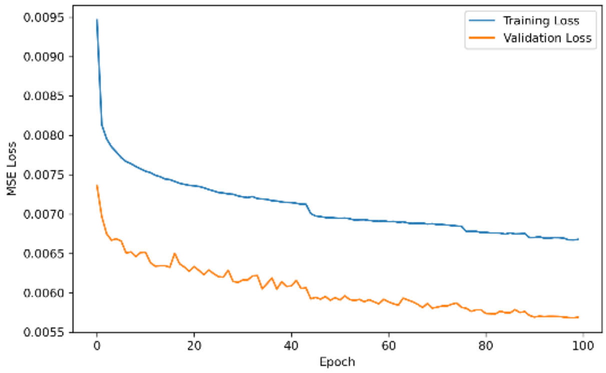


**Figure 6.** Training loss and validation loss curves of the improved model.

**Table 3.** Comparison of quantitative performance before and after adam algorithm improvement.

| Indicator | O-Adam | I-Adam | IA |
|---|---|---|---|
| CIT | 60epoch | 50epoch | 16.67% |
| OVL | 0.0058 | 0.0056 | 3.45% |
| SD | 0.00035 | 0.00025 | 28.57% |

From the various indicator data in Table 3, it can be seen that the improved Adam algorithm reduces the number of iterations required for the model to converge from about 60 in the original Adam to 50, thereby increasing the convergence efficiency of the model by 16.67%; The final validation loss was optimized from 0.0058 to 0.0056, reducing the error by approximately 3.45%. In addition, the standard deviation of the validation loss decreased from 0.00035 to 0.00025, and the fluctuation amplitude decreased by about 28.57%, indicating that the improved Adam significantly improved the stability of the model during training. By combining the loss curves in Figures 5 and 6 with the quantitative indicators in Table 3, it can be seen that the improved Adam algorithm further reduces validation loss and training fluctuations, significantly improving the convergence efficiency and generalization ability of the model. In the multi pollutant prediction task of this study, this improvement effectively solves the gradient conflict problem under multimodal feature input, providing a more stable and efficient optimization path for the model.

### 3.3 Evaluation Indicators

This study selected MAE and coefficient of determination $R^2$ as evaluation metrics, and MSE as the loss function. The specific formula for the loss function MSE is as follows:

$$L_1 = \frac{1}{C}\sum_{i=1}^{C}(y_i - \hat{y}_i)^2 \tag{22}$$

$$\text{Loss} = \sum_{p=1}^{P}\omega_t L_p \tag{23}$$

In the formula, $L_p$ represents the loss value of the $p$-th task, $P$ represents the number of tasks, $C$ represents the number of samples, $y_i$ represents the true value, $\hat{y}_i$ represents the predicted value, Loss represents the total loss value, and $\omega_t$ represents the weight of the task.

### 3.4 Comparative Experiment

To comprehensively evaluate the performance of the MLP-BiLSTM-MHAT model, this study selected four representative models in time series regression prediction for comparative experiments. Specifically, it includes:

- 1) **GRU and LSTM**: as classic benchmarks for recurrent neural networks, used to evaluate the model's basic ability to capture short-term dependencies;
- 2) **TCN**: Known for its causal dilation convolution structure, it can efficiently capture long-term dependencies and aims to test the effectiveness of the proposed model in capturing long-range information;
- 3) **Transformer**: relies on self-attention mechanism to globally model sequence dependency relationships, used to verify the performance of the proposed model in complex dependency patterns and computational efficiency.

By comparing with the above models, the aim is to systematically verify the comprehensive advantages of the MLP-BiLSTM-MHAT model from multiple dimensions such as basic recursive ability, long-term dependency modeling, and global dynamic capture. The specific prediction results are detailed in Table 4.

**Table 4.** Comparison of prediction results of five models.

| Model | RMSE | MAE | $R^2$ |
|---|---|---|---|
| GRU | 103.695 | 32.637 | 0.912 |
| LSTM | 102.365 | 31.175 | 0.924 |
| TCN | 101.824 | 30.827 | 0.930 |
| Transformer | 101.282 | 30.480 | 0.937 |
| MBM | 100.741 | 30.132 | 0.943 |

From Table 4, it can be seen that the MBM model performs the best in all evaluation metrics. Its RMSE value is 100.741, which is an average decrease of about 1.9% compared to other models, indicating that this model has higher prediction accuracy; The MAE value is 30.132, with an average decrease of 4.2%, indicating that the MBM model has stronger robustness in dealing with extreme or outlier values; At the same time, its $R^2$ value is 0.943, with an average improvement of about 1.8%, which can explain 94.3% of the data variation and demonstrate significantly improved ability to capture variation data.

In order to demonstrate the prediction performance more intuitively, Figures 7 to 16 further compared the prediction results and actual values of the MLP-BiLSTM-MHAT model with other models, visually verifying its advantages in fitting and stability.

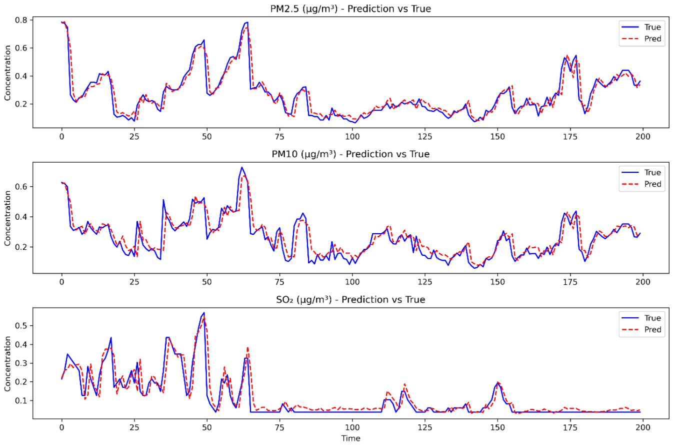


**Figure 7.** Comparison chart of PM2.5, PM10, $SO_2$ between MLP-BiLSTM-MHAT model and actual values.

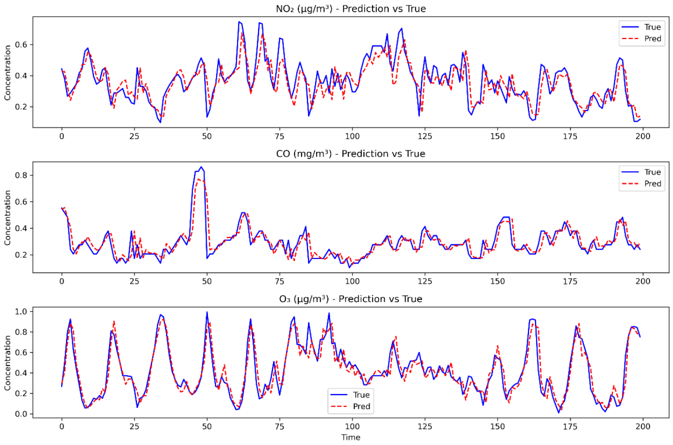


**Figure 8.** Comparison chart of $NO_2$, CO, $O_3$ between MLP-BiLSTM-MHAT model and actual values.

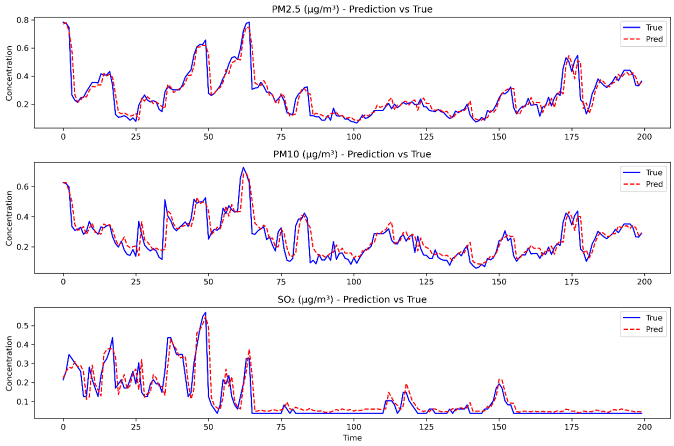


**Figure 9.** Comparison chart of PM2.5, PM10, $SO_2$ between GRU model and actual values.

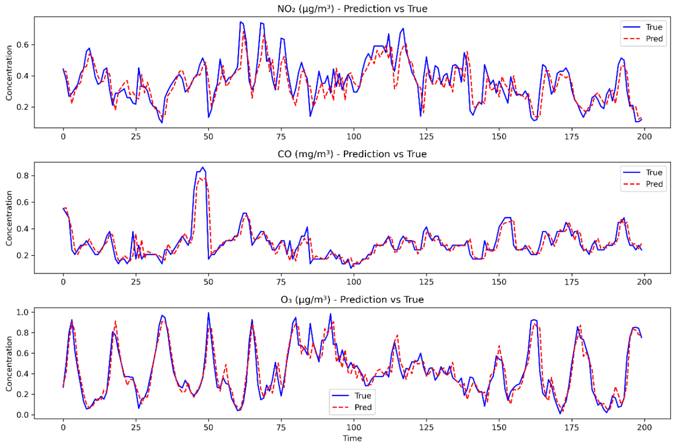


**Figure 10.** Comparison chart of $NO_2$, CO, $O_3$ between GRU model and actual values.

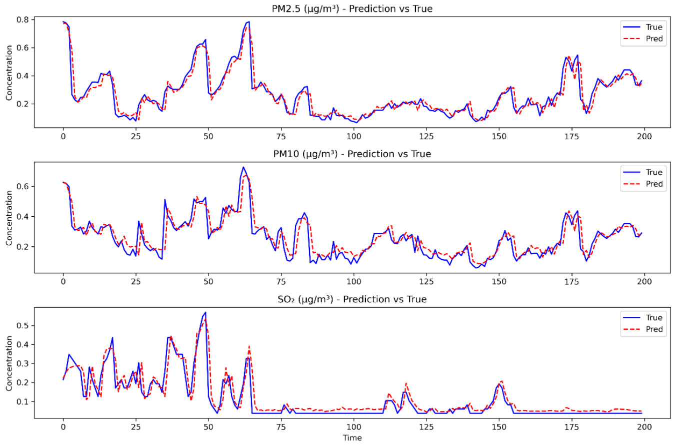


**Figure 11.** Comparison chart of PM2.5, PM10, $SO_2$ between LSTM model and actual values.

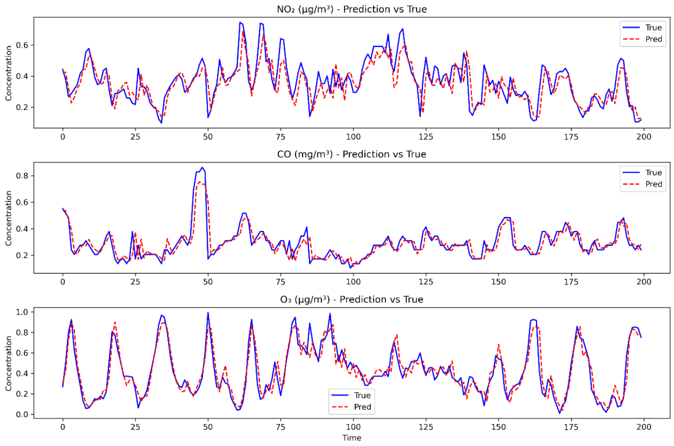


**Figure 12.** Comparison chart of $NO_2$, CO, $O_3$ between LSTM model and actual values.

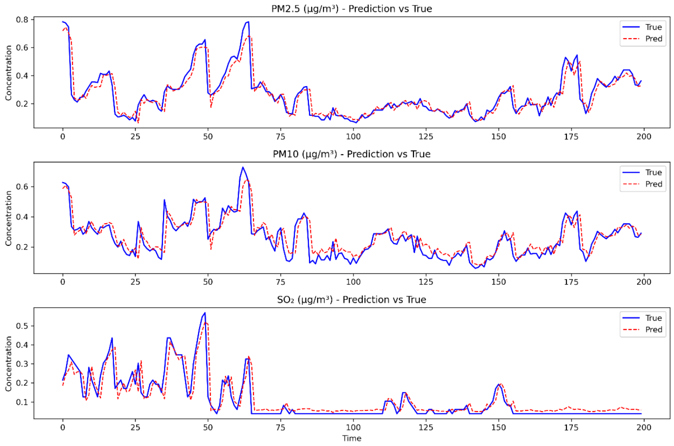


**Figure 13.** Comparison chart of PM2.5, PM10, $SO_2$ between TCN model and actual values.

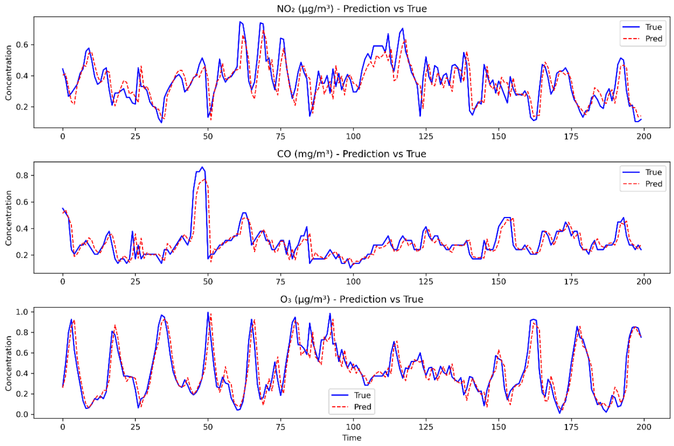


**Figure 14.** Comparison chart of $NO_2$, CO, $O_3$ between TCN model and actual values.

By comparing the predicted results of different air pollutant concentrations, it can be seen that in the prediction of PM2.5 and PM10, the MLP-BiLSTM-MHAT model can accurately capture the trends of peak and valley values, especially in the mutation area, the fitting effect is significantly better than other models; In the prediction of $SO_2$, the curve of the MLP-BiLSTM-MHAT model almost

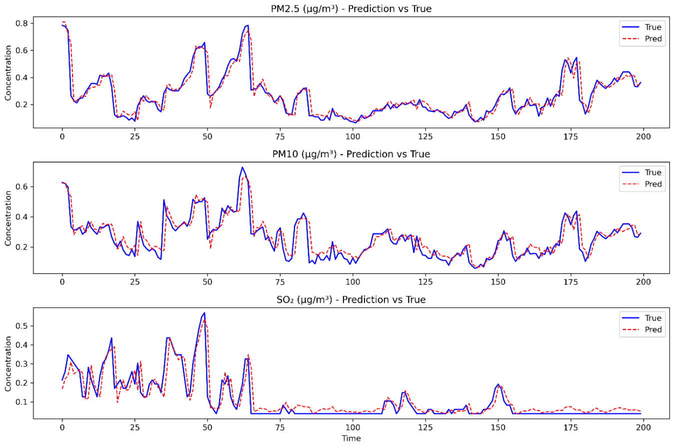


**Figure 15.** Comparison chart of PM2.5, PM10, $SO_2$ between Transformer model and actual values.

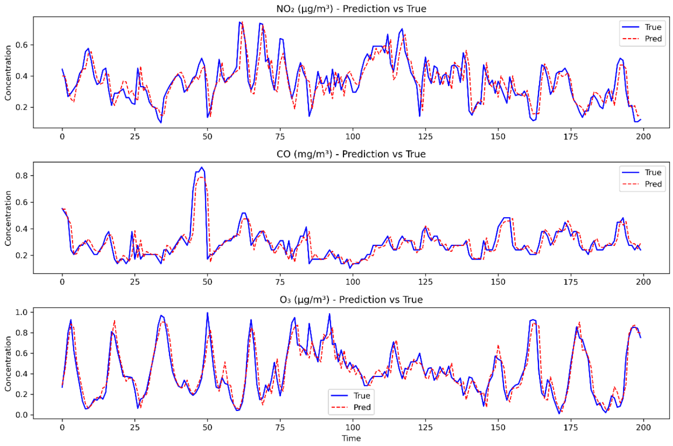


**Figure 16.** Comparison chart of $NO_2$, CO, $O_3$ between Transformer model and actual values.

coincides with the actual value, demonstrating its high-precision modeling ability for low concentration gas fluctuations; In the prediction of $NO_2$, all models have some degree of noise, but MLP-BiLSTM-MHAT is smoother and more accurate in fitting the trend of continuous fluctuations. In the prediction of CO and $O_3$, MLP-BiLSTM-MHAT also demonstrated good performance, not only effectively capturing their periodic features, but also significantly better fitting accuracy in the high concentration range than the other four models. By comparing the prediction results of different air pollutants, it can be seen that this method has higher accuracy and robustness in predicting air pollutant concentrations.

## 4 Conclusion

This article proposes a multimodal deep learning model that integrates MLP, BiLSTM, and multi-head attention mechanism, and applies it to predict air pollutant concentrations. By introducing memory units to improve the Adam optimization algorithm, the gradient conflict problem in the multimodal feature fusion process has been effectively alleviated, thereby enhancing the convergence efficiency of the model in complex time dependent and static correlated data. The experiment used RMSE, MAE and R2 as evaluation metrics, and compared the performance of the proposed model with multiple representative benchmark models. The results showed that the model exhibited higher accuracy and robustness in predicting various air pollutant concentrations, with an average RMSE decrease of 1.9%, an average MAE decrease of 4.2%, and an average R2 increase of 1.8%. The main contribution of this study is to provide an effective framework for collaborative prediction of multiple pollutants, and its performance advantages reflect that the model can capture the deep spatiotemporal evolution laws shared by different pollutants, which has good application value in the field of regional environmental air quality warning.

## Data Availability Statement

Data will be made available on request.

## Funding

This work was supported without any funding.

## Conflicts of Interest

The authors declare no conflicts of interest.

## AI Use Statement

The authors declare that no generative AI was used in the preparation of this manuscript.

## Ethical Approval and Consent to Participate

Not applicable.

## References

[1] Wang, S., Qiao, L., Fang, W., Jing, G., & Zhang, Y. (2022). Air Pollution Prediction Via Graph Attention Network and Gated Recurrent Unit. *Computers, Materials & Continua*, *73*(1). [CrossRef]

[2] Zhang, A., Qi, Q., Jiang, L., Zhou, F., & Wang, J. (2013). Population exposure to PM2. 5 in the urban area of Beijing. *PloS one*, *8*(5), e63486. [CrossRef]

[3] Kim, Y., & Radoias, V. (2022). Severe air pollution exposure and long-term health outcomes. *International Journal of Environmental Research and Public Health*, *19*(21), 14019. [CrossRef]

[4] Bekkar, A., Hssina, B., Douzi, S., & Douzi, K. (2021). Air-pollution prediction in smart city, deep learning approach. *Journal of big Data*, *8*(1), 161. [CrossRef]

[5] Qin, D., Yu, J., Zou, G., Yong, R., Zhao, Q., & Zhang, B. (2019). A novel combined prediction scheme based on CNN and LSTM for urban PM 2.5 concentration. Ieee Access, 7, 20050-20059. [CrossRef]

[6] Zhou, G., Xu, J., Xie, Y., Chang, L., Gao, W., Gu, Y., & Zhou, J. (2017). Numerical air quality forecasting over eastern China: An operational application of WRF-Chem. *Atmospheric Environment*, *153*, 94-108. [CrossRef]

[7] Hinojosa-Baliño, I., Infante-Vázquez, O., & Vallejo, M. (2019). Distribution of PM2. 5 air pollution in Mexico City: Spatial analysis with land-use regression model. *Applied sciences*, *9*(14), 2936. [CrossRef]

[8] Sharma, V., Ghosh, S., Mishra, V. N., & Kumar, P. (2025). Spatio-temporal Variations and Forecast of PM2. 5 concentration around selected Satellite Cities of Delhi, India using ARIMA model. *Physics and Chemistry of the Earth, Parts A/B/C*, *138*, 103849. [CrossRef]

[9] Amin, R., Salan, M. S. A., & Hossain, M. M. (2024). Measuring the impact of responsible factors on CO2 emission using generalized additive model (GAM). *Heliyon*, *10*(4). [CrossRef]

[10] Gourav, Rekhi, J. K., Nagrath, P., & Jain, R. (2019). Forecasting air quality of delhi using arima model. In *Advances in Data Sciences, Security and Applications: Proceedings of ICDSSA 2019* (pp. 315-325). Singapore: Springer Singapore. [CrossRef]

[11] Cortina–Januchs, M. G., Quintanilla–Dominguez, J., Vega–Corona, A., & Andina, D. (2015). Development of a model for forecasting of PM10 concentrations in Salamanca, Mexico. *Atmospheric Pollution Research*, *6*(4), 626-634. [CrossRef]

[12] Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, *2*(5), 359-366. [CrossRef]

[13] Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural networks*, *18*(5-6), 602-610. [CrossRef]

[14] Yang, H., Xiao, K., Xiang, X., Wang, X., Wang, X., Du, Y., ... & Yang, F. (2025). Prediction of on-road CO2 emissions with high spatio-temporal resolution implementing multilayer perceptron. *Atmospheric Environment: X*, 100368. [CrossRef]

[15] Aamir, M., Bhatti, M. A., Bazai, S. U., Marjan, S., Mirza, A. M., Wahid, A., ... & Bhatti, U. A. (2022). Predicting the environmental change of carbon emission patterns in South Asia: a deep learning approach using BiLSTM. *Atmosphere*, *13*(12), 2011. [CrossRef]

[16] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems, 30*.

[17] Jin, X., Sun, T., Chen, W., Ma, H., Wang, Y., & Zheng, Y. (2024). Parameter adaptive non-model-based state estimation combining attention mechanism and LSTM. *ICCK Transactions on Intelligent Systematics*, *1*(1), 40-48. [CrossRef]

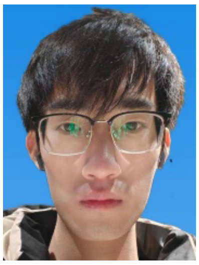
[18] Dai, Z., Yang, Z., Yang, Y., Carbonell, J. G., Le, Q., & Salakhutdinov, R. (2019, July). Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 2978-2988). [CrossRef]

[19] Yu, F., & Koltun, V. (2015). Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.

[20] Li, W., & Jiang, X. (2023). Prediction of air pollutant concentrations based on TCN-BiLSTM-DMAttention with STL decomposition. *Scientific Reports*, *13*(1), 4665. [CrossRef]

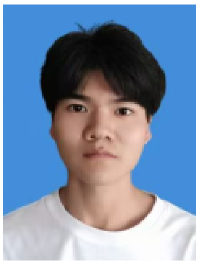
[21] Jabeen, S., Li, X., Amin, M. S., Bourahla, O., Li, S., & Jabbar, A. (2023). A review on methods and applications in multimodal deep learning. *ACM Transactions on Multimedia Computing, Communications and Applications*, *19*(2s), 1-41. [CrossRef]

[22] Kinga, D., & Adam, J. B. (2015, May). A method for stochastic optimization. In *International conference on learning representations* (ICLR) (Vol. 5, No. 6).



**Chenbin Gu**, from Huai'an City, Jiangsu Province, China, obtained a bachelor's degree in Electronic Science and Technology from Huaiyin Institute of Technology. He is currently a master's student at Huaiyin Institute of Technology, with a research focus on Traffic Information Engineering and Control. (Email: gcb31830127@126.com.)

**Xiaoqi Yin**, corresponding author, is a professor and master's supervisor in Electronic Science and Technology from Huaiyin Institute of Technology.She has been engaged in research on signal processing and measurement technology for a long time. (Email: hy_xuebao2009@126.com)



**Xuejun Li**, from Zhoukou' City, Henan Province, China, obtained a bachelor's degree in Electronic Science and Technology from Huaiyin Institute of Technology. He is currently a master's student at Huaiyin Institute of Technology, with a research focus on Traffic Information Engineering and Control. (Email: lixuejun1013@163.com)