Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th June 2025. Vol. 4. No. 3; e-ISSN: 2808-4519

Predicting CO levels using LSTM and Rolling-Features

Afifa Salsabila[1]*, Anggraini Puspita Sari[2], Achmad Junaidi[3]

[1,2,3]Universitas Pembangunan Nasional "Veteran" Jawa Timur
afifasals@gmail.com[1*]

Abstract

Accurately predicting carbon monoxide (CO) levels is essential for effective environmental monitoring and safeguarding public health. This research investigates the use of Long Short-Term Memory (LSTM) networks for forecasting CO concentrations specifically evaluating how different learning rates influence model performance. The study aimed to assess the effects of adjusting the learning rate to 0.001, 0.0001, and 0.0005 on the model's accuracy and rate of convergence. A dataset of CO measurements was utilized with feature engineering applied to include lag-based and rolling window features. Results indicated that a learning rate of 0.001 produced the most accurate predictions, achieving the lowest error metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Conversely, smaller learning rates resulted in higher error rates, reflecting slower convergence and less accurate predictions. These findings underscore the importance of selecting the correct learning rate for optimal model performance and suggest that future studies could further investigate learning rate optimization and integrate additional data to improve prediction outcomes.

Keywords: *Carbon Monoxide, LSTM, Prediction, Rolling Features, Rolling Mean*

1. Introduction

Forecasting the concentration of environmental pollutants such as carbon monoxide (CO) is essential for protecting public health and supporting sustainable urban planning. Accurate CO level forecasting can help reduce health risks and enable more effective policy interventions [1]. Long Short-Term Memory (LSTM) networks, a type of Recurrent Neural Network (RNN), are well-suited for time series forecasting due to their ability to capture long-term dependencies in sequential data [2]. Recent advancements have integrated rolling-based features, such as rolling averages and rolling standard deviations, with LSTM networks, enhancing forecasting accuracy by improving the model's understanding of temporal dependencies and variance in the data. This study models CO concentrations using an LSTM network trained with a dataset that includes not only historical CO values but also features derived from rolling statistics, such as rolling average and rolling standard deviation [3]. These features help the model capture the dynamics of CO levels over time, enabling it to learn short-term fluctuations and long-term trends. Given the complex temporal patterns of CO, which are influenced by various environmental factors such as weather, traffic, and industrial emissions, the inclusion of lag features and rolling statistics significantly improves the model's forecasting accuracy.

LSTM networks are highly effective in this context due to their ability to handle data with different time intervals, which is characteristic of meteorological and air quality datasets. This characteristic allows the model to handle irregular time steps, which are common in real-world datasets [4]. Additionally, the use of MinMaxScaler for feature normalization ensures that the model can handle a wide range of input values and improves its stability during training. To further optimize the model, techniques such as early stopping and learning rate reduction are applied to avoid overfitting and improve training efficiency. These methods are crucial when working with real-world environmental data, which often contains noise, outliers, and missing values. The application of these techniques ensures that the model remains generalizable and reliable in real-world scenarios.

Evaluation metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) are used to assess model performance, providing clear indications of prediction accuracy and reliability [5]. These metrics are crucial for determining the model's effectiveness in practical applications, such as air quality monitoring and public health interventions. By focusing on improving forecasting precision, this research contributes to ongoing efforts to create more reliable and efficient tools for environmental monitoring. Additionally, the integration of rolling statistics and LSTM models may pave the way for future research to incorporate additional data sources, such as real-time sensor data, to further enhance prediction accuracy and expand the scope of air quality forecasting systems.

2. Methods

This section describes the methodology used to predict carbon monoxide (CO) concentrations using a Long Short-Term Memory (LSTM) network, incorporating rolling-based features. The main objective of this study is to develop a reliable model capable of accurately predicting CO concentrations, a critical task for air quality monitoring and public health interventions. This methodology encompasses data preprocessing, feature engineering, model architecture, training, and evaluation. Each step is carefully designed to maximize model performance, enhance predictive capabilities, and ensure model generalization.

2.1. Pre-processing Data

The dataset used in this study consists of CO concentration measurements collected over a specific period of time. Preprocessing steps are essential to transform raw data into a form suitable for input into the LSTM model. This data set contains a date-time column (DATE). To make it suitable for time series analysis, the date column is converted to a date-time format, enabling time-based indexing and easier handling of time-dependent data. Missing data in CO concentration measurements is handled using forward fill and backward fill techniques. Forward fill propagates the last valid observation forward, while backward fill fills in missing values with the next valid observation. This method ensures data continuity, which is critical for time series models. To stabilize variance and make the CO distribution more suitable for analysis, a logarithmic transformation is applied. The logarithmic transformation is given by Equation (1). This transformation helps reduce the impact of outliers and makes the data distribution more symmetric, improving model performance.

$$CO_{log} = \log(CO + 1) \tag{1}$$

2.2. Feature Engineering

Feature engineering is a critical step that aims to extract relevant information from raw data to improve the predictive power of the model [6]. Lag features are created by shifting the CO concentration values to the previous time step. These lag features represent CO concentrations from the past, capturing the temporal dependence between past and future values. Lag values can be represented mathematically in Equation (2).

$$CO_{Lag\ t} = CO(t-k) \tag{2}$$

The symbol k represents the lag step, and t denotes the current time step. For example, for a lag of 1, $CO_{Lag\ t}1$ represents the CO concentration value from the previous time step. To capture local trends and fluctuations in the data, two rolling-based features are created: rolling mean and rolling standard deviation. These features are calculated over a 5-step time window. The moving average is calculated using Equation (3), while the moving standard deviation is given by Equation (4).

$$Rolling\ Mean_t = \frac{1}{\omega} \sum_{i=t-\omega+1}^{t} CO(i) \tag{3}$$

The symbol ω represents the window size. The moving average and moving standard deviation capture the average and variability of CO concentrations over the specified window. These features help the model understand short-term trends and variations in the data, which are important for predicting future concentrations.

$$Rolling\ Std_t = \sqrt{\frac{1}{\omega} \sum_{i=t-\omega+1}^{t} (CO(i) - Rolling\ Mean_t)^2} \tag{4}$$

2.3. Model Architecture

This forecasting model is based on a Long Short-Term Memory (LSTM) network, a type of recurrent neural network (RNN) that is well suited for time series forecasting. LSTM models are capable of capturing long-term dependencies in sequential data, which is essential for forecasting CO concentrations based on historical data [7]. The data is converted into sequences using a sliding window approach, where each window represents a specific time range. For each sequence, the model uses the last 7 days of data to predict the CO concentration on the 8th day. This is done using the create_time_sequence function, which generates a sequence from the processed data. The LSTM network is constructed using two LSTM layers, each followed by a dropout layer to prevent overfitting. The first LSTM layer consists of 32 units and uses ReLU activation with return_sequences=True to pass the sequence to the next LSTM layer. The second LSTM layer also has 32 units, but with return_sequences=False to produce a single prediction. Finally, a dense output layer with one unit is added to predict the CO concentration at the next time step. The model is compiled using the Adam optimizer and the mean squared error (MSE) loss function.

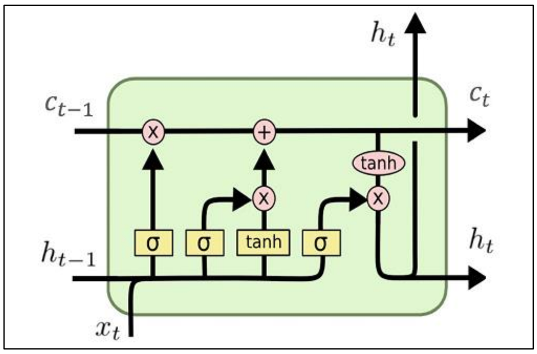

Fig. 1: LSTM Model Architecture

2.4. Model Training and Evaluation

The model is trained using a portion of the dataset, while the rest is set aside for testing. The training data is used to optimize the model weights, while the test data is used to evaluate the model's generalization performance. The model is trained using the mean squared error (MSE) loss function, which measures the average squared difference between the predicted and actual CO concentrations. The Adam optimizer is used to minimize the MSE loss function. The optimizer adjusts the model weights to improve its performance, with the learning rate as a key hyperparameter. To prevent overfitting, an early stopping implementation is applied. If the validation loss does not improve after a specified number of epochs, training is stopped, and the model with the best validation performance is restored. After training, the model's performance is evaluated using several key metrics, first the Mean Squared Error (MSE). MSE is used to measure the average squared difference between the predicted and actual CO concentrations. Next, the Root Mean Squared Error (RMSE) is the square root of MSE, providing an indication of the model's prediction error in units consistent with the target variable. Third, the Mean Absolute Error (MAE) measures the average absolute difference between the predicted and actual CO concentrations. Finally, the Mean Absolute Percentage Error (MAPE) measures the percentage difference between the predicted and actual values. To assess the model's generalization ability, predictions on the test set are compared to the actual CO concentrations. MSE, RMSE, MAE, and MAPE are calculated to measure the accuracy and performance of the model. These values are reported to provide insight into the model's predictive ability. Additionally, loss and MAPE plots are generated to visually examine training and validation loss throughout the epochs, providing insight into the training process and potential overfitting.

3. Results

The purpose of the results section is to present the data from this study clearly and objectively so that other researchers can draw their own conclusions and fully understand the basis for those conclusions. This section presents model performance metrics and visualizations that illustrate the results of the LSTM-based CO concentration forecasting model. This section presents data from the CO concentration forecasting study using the LSTM model. The results are organized into different scenarios, where the model is tested using three different learning rates: 0.001, 0.0001, and 0.0005. The metrics for each learning rate are presented separately, allowing for a comparison of model performance between learning rates. All results, including performance metrics and visualizations, are reported as raw data without interpretation.

3.1. Learning Rate 0,001

The performance of the model trained with a learning rate of 0.001 is assessed using loss curves and key metrics such as MSE, RMSE, MAE, and MAPE. The results from the training process, along with corresponding visualizations, provide a comprehensive view of the model's accuracy and the impact of the learning rate on its behavior. As shown in Figure 2, the performance metrics for the learning rate of 0.001 indicate that the model made accurate predictions overall. The Mean Squared Error (MSE) was computed to be 0.0013, which is a low value, signifying that the difference between the predicted and actual CO concentrations is minimal. This suggests that the model effectively fits the data and produces precise predictions. The Root Mean Squared Error (RMSE), which measures the magnitude of prediction errors, was found to be 0.0366, a relatively small value confirming that the model's predictions are close to the actual CO concentrations. Additionally, the Mean Absolute Error (MAE) was 0.0286, indicating that the model's predictions are off by just 0.0286 units of CO. The Mean Absolute Percentage Error (MAPE) was calculated at 4.29%, demonstrating that the model's predictions deviate by only 4.29% from the true CO values. This low MAPE reinforces the accuracy of the model's predictions, particularly in terms of relative error, supporting the efficiency of the model trained with this learning rate.

```
Mean Squared Error (MSE): 0.0013402616300714155
Root Mean Squared Error (RMSE): 0.03660958385547991
Mean Absolute Error (MAE): 0.02860419853405389
MAPE: 4.28636347307212
```

Fig. 2: Metric Evaluation Results with 0,001 Learning Rate

Figure 3 presents the training and validation loss curves throughout the training procedure. Initially, the training loss is relatively high at 0.2327 in the first epoch, but it drops rapidly in the following epochs, reflecting the model's fast learning phase. This sharp decline can be attributed to the large learning rate of 0.001, allowing the model to quickly adapt to the data. After the initial drop, the loss continues to decrease more gradually, leveling off around epoch 15. The validation loss follows a similar trend, with a small gap between the training and validation losses, indicating that the model generalizes well to new data. By epoch 32, the validation loss reaches about 0.0013, showing that the model has effectively minimized the error, and further training yields minimal improvements in loss.

Fig. 3: Training and Validation Loss with 0,001 Learning Rate

Figure 4 compares the predicted and actual CO concentrations for the test set. The blue line represents the actual CO values, while the red line shows the predicted values. The graph illustrates that the model's predictions closely follow the actual CO concentrations, especially during periods of stability. However, during more volatile periods, the predicted values tend to slightly lag or deviate from the actual values. This behavior is expected, as time series models like LSTM tend to perform well when predicting steady patterns but may face challenges in handling high volatility or noise in the data.

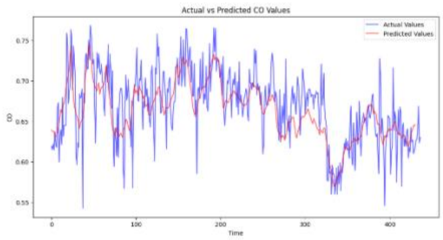

Fig. 4: Actual vs Predicted Values with 0,001 Learning Rate

3.2. Learning Rate 0,0001

The performance metrics of the model trained with a learning rate of 0.0001 indicate that while the model performs reasonably well, there are some noticeable differences compared to the model trained with a learning rate of 0.001. As shown in Figure 5, the Mean Squared Error (MSE) was calculated to be 0.00198, which suggests that the model's predictions have a slightly larger average squared deviation from the actual CO concentrations. The Root Mean Squared Error (RMSE) was found to be 0.0445, indicating that the average magnitude of error in the model's predictions is larger compared to the model trained with the 0.001 learning rate. The Mean Absolute Error (MAE) was 0.0348, meaning that the model's predictions deviate by 0.0348 units of CO, which is slightly higher than the MAE observed for the model trained with a learning rate of 0.001. Lastly, the Mean Absolute Percentage Error (MAPE) was 5.22%, indicating that the model's predictions deviate by an average of 5.22% from the actual CO values.

```
Mean Squared Error (MSE): 0.001980307309528156
Root Mean Squared Error (RMSE): 0.04450064392262382
Mean Absolute Error (MAE): 0.03481501888021118
MAPE: 5.218404813217303
```

Fig. 5: Metric Evaluation Results with 0,0001 Learning Rate

Figure 6 shows the training and validation loss curves throughout the training process. In the initial epochs, both training and validation losses are relatively high, with the training loss at 0.4417 and the validation loss at 0.3516. As the training progresses, the training loss decreases significantly, demonstrating the model's ability to adapt to the data. However, compared to the model trained with a higher learning rate (0.001), the decrease in loss is slower in the initial epochs, which can be attributed to the smaller step size associated with the lower learning rate. By epoch 36, the training loss approaches a value of 0.0123, and the validation loss stabilizes at 0.0020. The small gap between the training and validation losses suggests that the model generalizes well and is not overfitting to the training data.

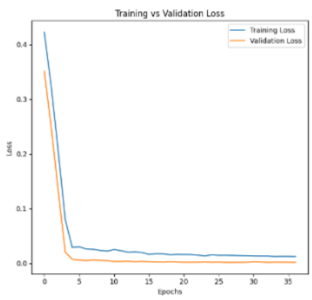

Fig. 6: Training and Validation Loss with 0,0001 Learning Rate

Figure 7 compares the predicted and actual CO concentrations for the test set. The blue line represents the actual CO concentrations, while the red line represents the predicted values. The predictions closely follow the actual values, with only slight deviations in periods with high fluctuations in CO levels. This pattern suggests that while the model effectively captures steady trends, it struggles with predicting more volatile fluctuations, which is common in time series models trained with smaller learning rates.

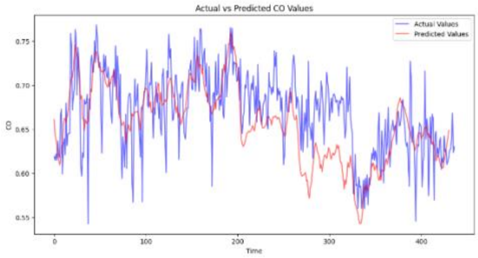

Fig. 7: Actual vs Predicted Values with 0,0001 Learning Rate

3.3. Learning Rate 0,0005

As illustrated in Figure 8, the model's performance metrics for the learning rate of 0.0005 demonstrate that the model performs effectively. The Mean Squared Error (MSE) was computed to be 0.00139, indicating that the average squared difference between the predicted and actual CO concentrations is relatively small. The Root Mean Squared Error (RMSE) for this model is 0.0373, reflecting the average magnitude of error in the predictions. The Mean Absolute Error (MAE) was found to be 0.0287, which means that, on average, the model's predictions deviate by 0.0287 units of CO. Finally, the Mean Absolute Percentage Error (MAPE) was calculated at 4.31%, indicating that, on average, the model's predictions deviate by 4.31% from the actual CO concentrations.

```
Mean Squared Error (MSE): 0.001392864179542414
Root Mean Squared Error (RMSE): 0.037321095636950616
Mean Absolute Error (MAE): 0.02866433819842473
MAPE: 4.3142997386737285
```

Fig. 8: Metric Evaluation Results with 0,0005 Learning Rate

Figure 9 displays the training and validation loss curves for the model with a learning rate of 0.0005. At the start, the training loss is 0.3176 in the first epoch, followed by a sharp decline, signifying that the model begins learning rapidly. The loss continues to decrease gradually, reaching approximately 0.0093 at epoch 35. The validation loss, represented by the orange line, follows a similar pattern, starting at 0.3516 and decreasing progressively. The gap between the training and validation losses remains relatively narrow, suggesting that the model generalizes well and does not overfit, despite the loss decrease being more gradual compared to higher learning rate scenarios. By the end of the training, both losses stabilize around the same value, implying that additional training would not lead to substantial improvements in model performance.

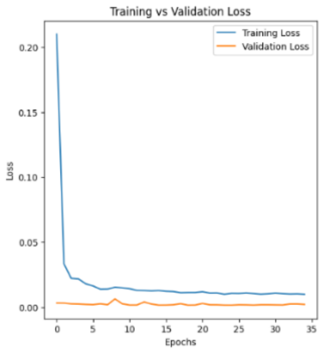

Fig. 9: Training and Validation Loss with 0,0005 Learning Rate

Figure 10 presents the comparison between the predicted and actual CO concentrations for the test set. The blue line represents the actual CO concentrations, while the red line depicts the predicted values. The predictions closely follow the actual CO concentrations, capturing the overall trend. Although there is some deviation during periods of high fluctuation, the model still manages to follow the general trend of the data quite well. This pattern is typical for time series models like LSTM, which tend to perform well with steady trends but may struggle with periods of rapid fluctuations or high volatility in the data.

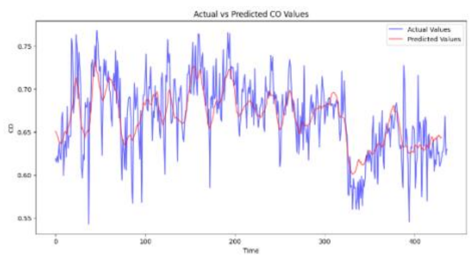

Fig. 10: Actual vs Predicted Values with 0,0005 Learning Rate

4. Conclusion

This research highlights the effectiveness of Long Short-Term Memory (LSTM) networks for predicting carbon monoxide (CO) concentrations, emphasizing the influence of various learning rates on model performance. The findings reveal that a learning rate of 0.001 resulted in the best performance, showing the lowest error metrics compared to smaller learning rates. The model trained with a learning rate of 0.0005, while slightly less precise, still produced reasonable predictions. The key insight is that the learning rate significantly affects the model’s ability to converge and generalize to new data. Although the study achieved promising outcomes, there is room for improvement. The model could be further refined by experimenting with more sophisticated learning rate schedules or by incorporating additional features, such as external data sources or more detailed CO measurements. Future research could also focus on enhancing the model's performance in handling periods of high volatility in CO concentrations. Despite these areas for development, the study lays a strong foundation for future advancements in air quality forecasting and demonstrates the potential for LSTM models in real-time environmental monitoring systems.

References

[1] G. Savioli *et al.*, “Carbon Monoxide Poisoning: From Occupational Health to Emergency Medicine,” May 01, 2024, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/jcm13092466.

[2] S. Yuan, C. Wang, B. Mu, F. Zhou, and W. Duan, “Typhoon intensity forecasting based on LSTM using the rolling forecast method,” *Algorithms*, vol. 14, no. 3, 2021, doi: 10.3390/a14030083.

[3] Y. Liu, H. Wang, W. Feng, and H. Huang, “Short term real-time rolling forecast of urban river water levels based on lstm: A case study in Fuzhou city, China,” *Int J Environ Res Public Health*, vol. 18, no. 17, Sep. 2021, doi: 10.3390/ijerph18179287.

[4] A. Yadav, C. K. Jha, and A. Sharan, “Optimizing LSTM for time series prediction in Indian stock market,” in *Procedia Computer Science*, Elsevier B.V., 2020, pp. 2091–2100. doi: 10.1016/j.procs.2020.03.257.

[5] K. Park, Y. Choi, W. J. Choi, H. Y. Ryu, and H. Kim, “LSTM-Based Battery Remaining Useful Life Prediction with Multi-Channel Charging Profiles,” *IEEE Access*, vol. 8, pp. 20786–20798, 2020, doi: 10.1109/ACCESS.2020.2968939.

[6] A. Moghar and M. Hamiche, “Stock Market Prediction Using LSTM Recurrent Neural Network,” in *Procedia Computer Science*, Elsevier B.V., 2020, pp. 1168–1173. doi: 10.1016/j.procs.2020.03.049.

[7] D. Seng, Q. Zhang, X. Zhang, G. Chen, and X. Chen, “Spatiotemporal prediction of air quality based on LSTM neural network,” *Alexandria Engineering Journal*, vol. 60, no. 2, pp. 2021–2032, Apr. 2021, doi: 10.1016/j.aej.2020.12.009.