Time Series Analysis: Application of LSTM model in predicting PM 2.5 concentration in Beijing

Rui Yang,

School of Cyber Science and Technology, Beihang University, China

Abstract. Air pollution forecasting for public health and policy-making has a critical importance, this paper employs a Long Short-Term Memory (LSTM) model to perform in-depth prediction of PM2.5 concentrations measured at the U.S. Embassy in Beijing, outperforming regular forecasting approaches. In the LSTM model, the research examines a very detailed hourly dataset and beats regular forecasting approaches. A key finding is the model's ability to effectively generalize from historical data to predict future air quality trends, with its adeptness at handling time-dependent relationships. This research emphasizes the importance of LSTM in air pollution prediction and management in environmental science as it provides an effective means for planning and making decisions on air quality management. This research is of great importance in providing a groundwork for further enhancement of prediction modeling. By offering a more reliable and sophisticated picture of air quality variations, this study addresses the current problem about how urban air pollution control could be improved in the city.

1 Introduction

The acceleration of industrialization and urbanization around the world has brought the economic prosperity, but it has also brought environmental problems, especially air pollution. Air pollution is mainly caused by fine particulate matter (PM 2.5). PM 2.5 has become the focus of its potential risk to respiratory and cardiovascular systems. PM2.5 has a complex source, including traffic exhaust, industrial emission, coal burning pollution and natural factor such as sandstorm [1]. Beijing, as the capital city of China, represents China's political and cultural center, its air quality not only related to citizens, but also affects China's international image and implement of sustainable development. In order to better monitor and respond to this problem, the U.S. Embassy in Beijing has been publishing the Air Quality Index (AQI) since 2008, which has become a vital source of information for monitoring PM2.5 levels. The Chinese government has implemented a number of measures to combat air pollution, including limiting vehicle use, closing polluting factories, and transitioning to clean energy. However, affected by multiple factors, these measures have not fully solved the air pollution problem in the short term. Air quality

Corresponding author: 22371418@buaa.edu.cn

in Beijing is still affected by seasonal and other complex factors, which makes the fluctuation of air pollution still a significant problem. Accurate PM2.5 predictions are essential for developing effective air quality management strategies. The result can help the government and relevant departments take measures in advance to reduce the occurrence of pollution incidents. Precise PM2.5 prediction can also be used as a basis for informed decisions of daily activities and travel plans.

However, the traditional air quality prediction method has some limitations, and it is difficult to accurately capture the dynamic change and long-term trend of PM2.5 concentration. To overcome these challenges, studies in recent years have discovered advanced machine learning methods, such as Long Short-Term Memory (LSTM), to improve the accuracy of the prediction. LSTM can effectively deal with the long-term dependency problem occurred in time series data. Through its unique gating mechanism, LSTM is able to quickly respond to and update new information while maintaining the memory of historical information, which gives LSTM a significant advantage in the field of air quality prediction. In recent years, LSTM has been extensively used in the area of air quality prediction. Recent studies have begun to consider factors such as weather conditions, fuel use, and traffic into air quality prediction models. These models with multiple factors, for example, hybrid LSTM and wavelet transform models (such as NLSTM) can predict air pollution levels more accurately [2]. Fine area modeling is also highly utilized by capturing local environmental features and special polluted sources in order to be more precise. In some researches, LSTM has also been combined with other neural networks such as GCN to increase the accuracy of prediction [3].

This paper uses LSTM for multivariate time series prediction to accomplish the forecast of air pollution levels at the U.S. Embassy in Beijing, China. By pre-processing the dataset and using MSE, MAE, RSME as metrics of the model, this LSTM can provide reference for future long-term forecasting research.

2 Data and Method

2.1 Data

The dataset is derived from the air quality dataset on kaggle [4]. This dataset is a well-measured collection of hourly observation, spanning a five-year period since 2010, capturing nuances of air quality and weather conditions at the U.S. embassy in Beijing, China. It is characterized by its high temporal resolution, offering a granular view into the hourly variations of PM2.5 concentration, which is a critical indicator of air quality.

The features of the dataset include a series of known meteorological variables that affect air quality, including: Dew point, provides information about the humidity of the atmosphere. Temperature, a basic measure of the thermal state of air. Atmospheric pressure, a key factor affecting air density and pollutant dispersal potential. Wind direction and speed, which are crucial for the transport and distribution of air pollutants. In addition, the dataset records the cumulative hours of snow and rain, variables that can significantly alter the dispersion and deposition of pollutants.

2.2 Pre-processing

The pre-processing of the air quality dataset is an intricate process, including several important steps to make sure the data is suitable for modeling. In this task, data pre-processing aims to ensure that the data is formatted, structured, and distributed in a

manner compatible with the input requirements of machine learning models, thereby enhancing the efficiency of model training and the accuracy of predictions.
Initially, the dataset is transformed from categorical variables to numerical format, which is crucial for compatibility with computational models that need numerical inputs. This transformation is achieved by a mapping technique that assigns a unique numerical identifier to each class of wind direction variables. For example, map function is used to transform the attributes of wind direction into numbers. NE, NW, SE, and cv are transformed into 0, 1, 2 and 3 correspondingly.

After the categorical data encoding, the temporal aspect of the dataset is highlighted by converting the date entries into datetime format, allowing temporal analysis and time series indexing. This transformation enables data points to be aligned along the time axis, which is essential for time-dependent modeling.

Another critical step in pre-processing pipeline is the normalization of the dataset. This is achieved by scaling the continuous variables to a uniform range, which is essential to neutralize the effects of different variable scales and to ensure that all features contribute equally to the model. To normalize the feature space and mitigate the impact of varying scales, the MinMaxScaler module is utilized [5]. Lists, encompassing the pollution level, dew point, temperature, atmospheric pressure, wind direction, wind speed, snow, and rain, are selected to be normalized. This selection is based on the relevance of these features to the prediction task at hand. The scaler is applied to training and testing datasets once it has been fitted to the training data, ensuring that the features are scaled to a range of 0 to 1, ensuring that the statistical distribution of the data is preserved while the input constraints required by the modeling algorithm are respected.

To enable the use of machine learning techniques, the structured data must be transformed into a format that can be used for numerical manipulation as the last stage of pre-processing. Specifically, Pandas DataFrame is transformed into a NumPy array before training the model because NumPy array is more flexible to manipulate [6]. Also, this transformation is done to simplify the data structure and thus optimize it for the computational efficiency required by advanced analysis methods.

2.3 Model

This article employs the Long Short-Term Memory (LSTM) paradigm. A particular type of recurrent neural network (RNN) called an LSTM network is able to recognize order dependency in sequence prediction issues. An LSTM network's core is the LSTM cell, which is made up of an input gate, an output gate, a forget gate, and a memory cell. These gates control information entering and leaving the cell, enabling it to accumulate and store pertinent data over time [7].

The memory cell's state, denoted as c_t , is updated at each time step based on the forget gate f_t , the input gate i_t , and the candidate value $\tilde{C}_t$ through the subsequent formulas:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (1)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (3)$$

$$\Delta C_t = f_t * c_{t-1} + i_t * \tilde{C}_t \quad (4)$$

$$c_t = \tanh(\Delta C_t) \quad (5)$$

In this case, σ stands for the sigmoid function, which outputs values ranging from 0 to 1, and x_t is the input at time step t. The weight matrices W_f, W_i, W_C and bias vectors b_f, b_i, b_C are parameters that are learned during training.

The output h_t of the LSTM cell at time steptis computed by:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (6)$$

$$h_t = o_t * \tanh(c_t) \quad (7)$$

The output h_t serves as a summary of the relevant information from the input sequence up to that point [8].

LSTM network is used in this realm of time series forecasting. Before training the model, the prepared dataset transforms to align with the requirements of LSTM network modeling. The dataset is divided into input-output pairs, with each input representing a series of previous observations and the output being the value to be predicted. This sequence-to-point transformation is pivotal for training the LSTM model to recognize patterns and make predictions based on the temporal dynamics inherent in the data.

The dataset is divided using a rolling window approach. This procedure results in a series of input vectors and scalar outputs, which are then encapsulated within arrays suitable for the LSTM model's architecture.

The LSTM model is adept at handling sequences through multiple LSTM layers, with the argument return_sequences set to True, which facilitates the capture of temporal dependencies and the preservation of information flow across layers. A dropout layer appears after the LSTM layers, which is integrated to introduce regularization, thereby reducing the model's reliance on any single input feature and enhancing its generalization capability. To prevent over-fitting, the training process incorporates early stopping, which monitors the validation loss and halts training when the performance plateaus or deteriorates, thus avoiding the model's excessive adaptation to the training data [9]. Additionally, a checkpoint is established to salvage the best-performing model configuration based on the validation loss, ensuring that the model's predictive accuracy is preserved. The training of the LSTM model is conducted through an iterative optimization process, where the model's parameters are adjusted to minimize the prediction error.

The selection of parameters is crucial for optimizing the training process, ensuring both efficiency and model generalization. The epochs parameter set as 150 allows for sufficient training iterations to capture complex temporal dynamics without succumbing to over-fitting. A batch size equals to 32 strikes a balance between computational tractability and the model's capacity to learn from data variations, promoting a stable and robust convergence. Setting validation_split to 0.1 sets aside a fraction of the training data for continuous performance evaluation, offering insights into the model's predictive capabilities on unseen data and enabling the early detection of over-fitting. The choice of setting shuffle as false respects the time series data's sequential integrity, which is paramount for an LSTM model to accurately learn and predict time-dependent patterns [10].

Upon completion of the training regimen, the model's performance is rigorously evaluated using the test dataset. The predictions generated by the model are then compared against the actual values to ascertain the model's predictive accuracy. Given the data's prior normalization, an essential final step involves applying an inverse transformation to the predictions. This technique translates the model's normalized output back into the initial scale, ensuring that predictions are interpretable within their empirical context.

3 Result

Several metrics are presented in order to assess the model's validation: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Squared Error (MSE). The average squared difference between the actual and anticipated values is measured by MSE. RMSE is the square root of MSE, as inferred from MSE. Error magnitude is given in the same

units as the data via RMSE, which facilitates interpretation. The average size of a group of forecasts' mistakes is assessed using MAE. It is the mean of the absolute discrepancies between the actual observation and the forecast over the test sample [11].

Table 1. Metrics for related models

Metric / Model	MSE	RMSE	MAE
LSTM	0.0045	0.0671	0.0417
Linear Regression	0.0095	0.0977	0.0758
MLP	0.0070	0.0838	0.0620
Decision Tree	0.0066	0.0810	0.0583
SVR	0.0090	0.0951	0.0794

From Table 1, the LSTM model exhibited an MSE of 0.0045, RMSE of 0.0671, and MAE of 0.0417, which are greatly lower than those of other models such as Linear Regression, MLP, Decision Tree, and SVR. These findings imply that the LSTM model is better suited to grasp the subtleties of the information on air quality. The LSTM model's predictions are more in line with the actual values, as seen by the decreased error metrics, which indicates a higher prediction accuracy when dealing with complex patterns.

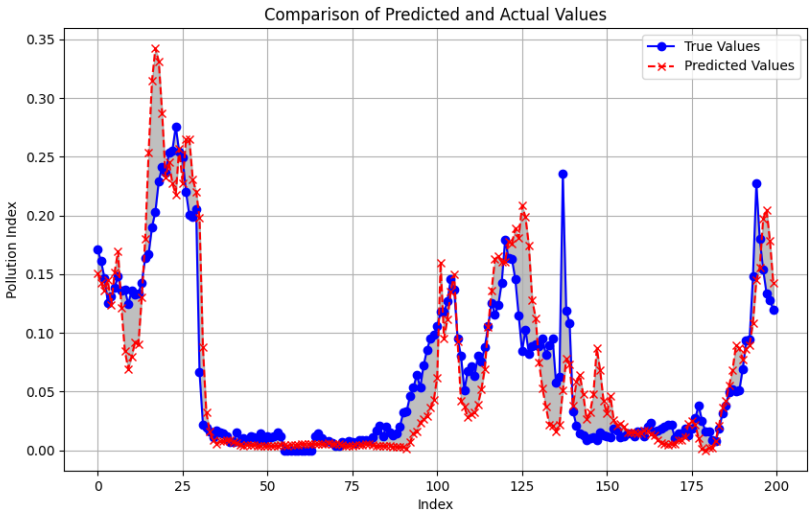

Fig. 1. Comparison of predicted and actual values (Photo/Picture credit : Original)

As shown in Fig. 1, the visualization of the model's predicted and actual values vividly illustrates the LSTM's ability of prediction. The closeness between the predicted and actual values validates the effectiveness of the model in transforming complex data sets into actionable insights. However, noting that any results which deviate from the equality matter. While the majority of the points are closely aligned, there may be some outliers or patterns of deviation that warrant further investigation. These deviations may stem from poor adaptation of the model to extreme weather conditions, or seasonal variations in the training data are not adequately captured.

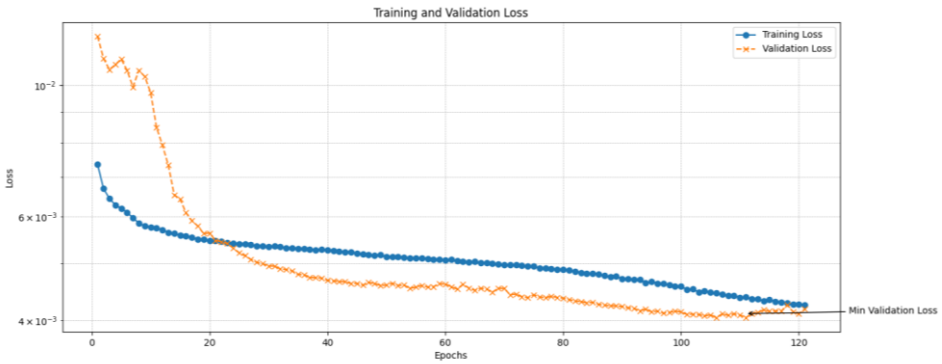

Fig. 2. Training and validation loss (Photo/Picture credit : Original)

Fig. 2 illustrates the model's learning process over epochs. The gradual decrease in both training and validation loss indicates effective learning, with the model converging to a minimum validation loss, suggesting good generalization. However, the slight gap between training and validation loss hints at the need for further regularization to prevent over-fitting.

The LSTM's lower error rates, can be attributed to its architecture that is specifically designed to address the challenges of sequential data. The information flow in LSTM cells is controlled by gates, which enable the model to retain significant patterns while discarding irrelevant input on a selective basis. This selective memory makes LSTM particularly well-suited for time-series prediction tasks such as air quality forecasting.

In conclusion, the lower error metrics of LSTM prove its effectiveness, highlighting its performance in handling time sequence data. The architecture's ability of LSTM is the prominent advantage over other models in dealing with sequences and long-term dependencies, making LSTM become the first choice when the missions of predicting future value based on historical data occurs. Future enhancements to the LSTM model could involve optimizing training duration and memory efficiency, employing advanced regularization techniques, and experimenting with different architectures. Additionally, integrating real-time data, leveraging ensemble methods, and incorporating external datasets could further improve predictive accuracy. Hyperparameter tuning and focusing on model interpretability will also be crucial for robust forecasting.

4 Conclusion

This paper uses LSTM to predict the concentration of PM 2.5, offering an hourly analysis of air quality at the U.S. Embassy in Beijing over a five-year period. The result of the study demonstrates that LSTM, with its intricate gating mechanisms, outperforms traditional models such as Linear Regression, MLP, Decision Tree, and SVR in terms of prediction accuracy proved by lower MSE, RMSE, and MAE values. Specifically, the MSE of the LSTM model is 0.0045, which is much lower than that of linear regression (0.0095), MLP (0.0070), decision tree (0.0066), and SVR (0.0090). Additionally, the LSTM model had the greatest performance out of all the models that were examined, with an RMSE of 0.0671 and an MAE of 0.0417. The ability to maintain information flow across layers and the robust handling of temporal data of LSTM have been pivotal in capturing the nuances of air quality fluctuations. The pre-processing step, including data encoding, normalization, and transformation, is crucial for preparing effective dataset for modeling. In comparison between prediction values and actual values, this model demonstrates a high degree of

alignment, suggesting its reliability in forecasting tasks. The gradual decrease in loss over epochs, underscores the model's capacity to generalize well to unseen data. The scope for future work lies in enhancing the LSTM model's predictive capabilities through refined training strategies and architectural improvements. Such advancements are anticipated to make a substantial contribution to the environmental science area, particularly in the realm of air quality assessment and mitigation strategies.

References

1. L. Yaolin, J. Zou, W. Yang, C.-Q. Li, A review of recent advances in research on PM2.5 in China. Int. J. Environ. Res. Public Health **15**, 438 (2018)
2. B. Liu, W. Chen, Z. Wang, S. Pouriyeh, M. Han, RAdam-DA-NLSTM: A Nested LSTM-Based Time Series Prediction Method for Human–Computer Intelligent Systems. Electronics **12**, 3084 (2023)
3. L. Yang, Z. Miao, T. Li, S. Tan, B. Wang, D. Li, Y. Liu, et al., LSTM-GCN based multidimensional parameter relationship analysis and prediction framework for system level experimental bench. Ann. Nucl. Energy **210**, 110890 (2025)
4. Air Pollution Forecasting - LSTM Multivariate, Kaggle, (2021) https://www.kaggle.com/datasets/rupakroy/lstm-datasets-multivariate-univariate
5. S. Patro, Normalization: A preprocessing stage. arXiv preprint arXiv:1503.06462 (2015)
6. C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, et al., Array programming with NumPy. Nature **585**, 357–362 (2020)
7. X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, W.-C. Woo, Convolutional LSTM network: A machine learning approach for precipitation nowcasting. Adv. Neural Inf. Process. Syst. **28** (2015)
8. R. C. Staudemeyer, E. R. Morris, Understanding LSTM--a tutorial into long short-term memory recurrent neural networks. arXiv preprint arXiv:1909.09586 (2019)
9. J. Brownlee, A gentle introduction to early stopping to avoid overtraining neural networks. Machine Learning Mastery **7** (2018)
10. J. Brownlee, How to tune LSTM hyperparameters with Keras for time series forecasting. (2017)
11. T. O. Hodson, Root mean square error (RMSE) or mean absolute error (MAE): When to use them or not. Geosci. Model Dev. Discuss. **2022**, 1–10 (2022)