Spatiotemporal prediction of O₃ concentration based on the KNN-Prophet-LSTM model

Biao Zhang a, Chao Song b, Ying Li b, Xuchu Jiang b,a

a School of Computer Science, Liaocheng University, Liaocheng, 252059, PR China
b Zhongnan University of Economics and Law, Wuhan, 430073, PR China

1. Introduction

In recent years, with the improvement of the economic level and the rapid development of cities, the concentration of ozone in the atmosphere has increased. As an oxidant, ozone is also an important factor in photochemical smog pollution. The increase in ozone concentration is harmful to human health, such as eye irritation, poor breathing, severe respiratory diseases and even suffocation. Plants in a high-concentration ozone environment will not perform photosynthesis normally, resulting in plant withering. Therefore, it is of great significance to accurately predict the ozone concentration in advance.

The existing ozone concentration prediction methods can be roughly divided into three directions. One is to optimize traditional machine learning algorithms and use various optimization methods to learn complex patterns and hidden features [1]. There are two main ideas: the Bayesian method and the kernel method. The former mainly learns the implicit patterns in the time series through model integration [2], and the latter is inclined to capture the relationship between hidden variables to improve the prediction accuracy [3, 4]. The second direction is based on the idea of signal decomposition. The development of frequency domain analysis methods is relatively mature. Applying signal decomposition methods, complex time series can be decomposed into several signals of different frequencies to simplify the prediction task, but the workload of such methods is relatively large, which need to be predicted separately and finally aggregated. [5, 6] The third direction is to use deep learning models to accurately analyze and predict ozone concentrations, taking into account spatial and temporal factors. The existing research is mainly divided into the improvement of the ex-
standing model and the data-driven prediction framework. Ma W et al. [7] proposed an optimized neural network model based on transfer learning, which first learned the intrinsic patterns of other pollutants (NO₂, PM₂.₅, etc.), and made accurate predictions of ozone concentration. Convolutional neural networks have great advantages in learning the spatial correlation of changes. Some scholars use convolutional neural networks to capture spatial change information [8, 9, 10] and then use sequence-to-sequence [11, 12] or encoding-decoding methods [13, 14] to simulate combined time series. The above models are all carried out on the premise of a large amount of data; otherwise, they may not converge. Therefore, in the case of small samples, the prediction results are not good. Some scholars have proposed data-driven prediction frameworks to perform feature engineering and model prediction by observing the characteristics of the data [15, 16].

At present, the pollutant concentration prediction methods for air monitoring stations mainly have the following problems: (1) ignoring the temporal correlation of station data: there is no long-term fine-grained prediction, and most of them are predictions for the next hour. (2) A deep learning model provides high accuracy, but its interpretability is poor. (3) Graph convolutional networks cannot directly extract long-term macro information in time series, which limits their application in the field of time series forecasting. In this paper, the KNN-Prophet-LSTM hybrid model is proposed to solve the above problems. The KNN algorithm is used to select the spatial correlation information of the target site, and the idea of time series decomposition is used to combine the traditional time series model with the deep learning model, which improves the prediction accuracy and increases the interpretability of the model from the two dimensions of time and space.

2. Models

2.1. The Prophet model

The Prophet is a time series prediction model which is opened by Facebook in 2017 [17]. The Prophet is popular for its flexibility and usability, which can automatically fill in missing values and has a very good prediction effect. The Prophet uses time series decomposition to predict and model the time series. The Prophet’s model constitutes are shown in Eq. (1):

\[ y(t) = g(t) + s(t) + h(t) + \epsilon \]  

where \( g(t) \) is a trend item, \( s(t) \) is a periodic term, \( h(t) \) is a holiday item, and \( \epsilon \) is a stochastic volatility term.

(1) Trend term: the trend term of the Prophet model is based on the improved logistic growth function to fit the aperiodic changes in the time series (Eq. (2)).

\[ g(t) = \frac{C(t)}{1 + \exp(-(k + a(t^T)\delta)(t - (m + a(t^T)\gamma)))} \]  

\[ a(t) = \begin{cases} 1, & t > s_j \\ 0, & \text{otherwise} \end{cases} \]  

where \( C \) denotes the capacity of the model, i.e., the saturation value of the growth, is the time \( t \). A function of \( k + a(t^T)\delta \) represents the growth rate of the model over time, \( m + a(t^T)\gamma \) represents an offset, \( s_j \) is a point of discontinuity at which the growth rate changes during a time series change, and \( \delta \) represents the amount of change in the growth rate at the mutation point.

(2) Periodic term: the Prophet model uses a Fourier series to model the periodicity of the time series (Eq. (3)).

\[ s(t) = \sum_{n=1}^{N} \left( a_n \sin \left( \frac{2\pi nt}{p} \right) + b_n \cos \left( \frac{2\pi nt}{p} \right) \right) \]  

where \( p \) represents a fixed period, and \( N \) represents the number of periods to be used in the model. \( a_n, b_n \) are the parameters to be estimated.

2.2. The LSTM model

Long short-term memory is an improved model based on the traditional recurrent neural network, which has good performance in time series prediction, and the LSTM has a more sophisticated information transmission mechanism. It can solve the problems of long-term memory deficiency, gradient disappearance or gradient expansion faced by Recurrent Neural Network (RNN) in the practical application process. Thus, the LSTM can deal with the long-term dependence problem in time series. The LSTM model structure is shown in Fig. 1.

The LSTM realizes the control and protection of the cell state by three structures of the input gate, output gate and forgetting gate. The input gate controls the inflow of information, the output gate controls the outflow of information, and the forgetting gate controls the memory unit to record the intensity of the historical cell state. The main functions of each state gate are as follows:

(1) Input gate: determines what information goes into the cell state and updates the cell state information. The sigmoid function determines the value to be updated, then the tanh function creates a new value vector, and finally, the latest cell state is updated (Eq. (5)).

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

\[ C_t = f_t C_{t-1} + i_t \times \tanh(W_c[h_{t-1}, x_t] + b_c) \]  

where \( W_i, W_c \) are weight vectors, and \( b_i, b_c \) are deviation vectors.

(2) Forget gate: by selectively processing the historical information, it is possible to determine which information in the cell state needs to be lost and which information needs to be retained. Enter as \( h_{t-1}, \) and \( x_{t-1} \), the forgetting gate is calculated by the sigmoid function (Eq. (6)).

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

where \( W_f \) is a weight vector and \( b_f \) is a deviation vector.

(3) Output gate: determine the information to be output. First, the sigmoid function is used to convert the output value into 0 and 1, where
1 represents output and 0 represents no output. Multiply the cell state with the obtained value to output the final information (Eq. (7)).

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

\[ h_t = o_t + \tanh(C_c) \]

(7)

where \( W_o \) is a weight vector and \( b_o \) is a deviation vector.

### 2.3. The KNN algorithm

The KNN algorithm is a supervised learning classification algorithm that is simple to implement and fast to train. It measures the similarity of two points in space by their distance. The smaller the distance is, the higher the similarity is. The test points are classified by the main category to which the nearest K points belong. The common distance measures are Euclidean distance, Mahalanobis distance, Manhattan distance and so on. The Euclidean distance is calculated in Eq. (8).

\[ d_{ij} = \left\| X_i - X_j \right\|_2 = \sqrt{\sum_{m=1}^{k} \left( x_i (m) - x_j (m) \right)^2} \]

(8)

where \( K \) denotes the dimension of the feature vector, \( x_i (m) \) and \( x_j (m) \) denote the value of the mth dimension, and \( m \) ranges from 1, 2, 3, …, \( K \).

### 2.4. The KNN-Prophet-LSTM prediction model

In the Prophet model, the time series fitting effect of historical data with a strong seasonal effect and several seasons is good, and it has strong robustness to missing data and trend changes. It can usually deal with outliers well, but the expression ability of the model is simple, so it is often unable to learn complex patterns when training the model. When the time series is not decomposed completely, it will lead to chaos in the remainder, and the Prophet model cannot fit the remainder well, which will reduce the prediction accuracy. As a deep learning model, the LSTM model can learn the potential relationship in a time series to fully extract effective information. If only the above model is used, the problem of not fully utilizing the spatial correlation characteristics is solved.

Therefore, the KNN algorithm is used to screen the adjacent spatial factors as an additional input to construct the KNN-Prophet-LSTM model to achieve a more accurate prediction of the O\(_3\) concentration. The specific prediction process is shown in Fig. 2.

1. Data preprocessing: before using the original data, we need to deal with the real samples and outliers. After statistics, there are no missing data; because the collection of data has gone through a series of strict audit processes, it is considered that the data are true and effective, and the data are not denoised. Because each pollution factor of a detection station has different dimensions and magnitudes, to reduce the error and speed up the model training, max-min normalization is applied to the data.

\[ x' = \frac{x_i - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} \]

(9)
where \( x_i \) is the original data, \( x_j \) is the normalized data, and \( x_{\text{max}} - x_{\text{min}} \) are the maximum and minimum values in the data respectively.

(2) The KNN algorithm is used to extract the spatial correlation characteristics of the target station. Let \( K = 1 \). In this paper, the Euclidean distance is used to measure the degree of correlation between the target station and the nearby stations. The smaller the distance is, the stronger the correlation. The KNN algorithm selection process is as follows.

1) According to the time step in the Prophet-LSTM model for a single site \( s_i \). Build \( O_s \) state matrix, there are \( m \) stations in the region. The state quantity of station \( i \) at time \( t \) in time step \( X_i \) and state matrix \( S_t \) in time \( t \) is shown in (Eq. (10)-Eq. (11)).

\[
X_t = \left( x_{i-1}, x_{i-2}, \ldots, x_{i-t}, x_{i+1}, x_{i+2}, \ldots \right)
\]

\[
S_t = \begin{bmatrix}
 x_{1-1} & x_{1-2} & \cdots & x_{1-t} & x_{1+1} & x_{1+2} & \cdots \\
 x_{2-1} & x_{2-2} & \cdots & x_{2-t} & x_{2+1} & x_{2+2} & \cdots \\
 \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots \\
 x_{m-1} & x_{m-2} & \cdots & x_{m-t} & x_{m+1} & x_{m+2} & \cdots \\
 \end{bmatrix}
\]

where \( x_{i} \) is the \( O_s \) concentration value at site \( u \) at time \( n \).

2) By calculating the Euclidean distances between the state quantity of the time step at moment \( t \) of the target station and other adjacent stations (\( m-1 \) in total), sorting, and selecting the data of the front \( K \) corresponding to the monitoring points as the space-related factors of the target station at moment \( t_i \), \( X_s \).

\[
X_s = \left( x_{i1}, x_{i2}, \ldots, x_{ip} \right)
\]

where \( x_{ij} \) is the site concentration value associated with the \( i \)th site of the target site at time \( t \).

(3) According to the historical pollution data of the target site, the Prophet-LSTM network is used to extract the temporal features. Enter as Target Site \( s_i \). Time step of the six pollutant concentrations of the verse, \( t \) moment input data \( \text{input}_{t} \). As shown in Eq. (12).

\[
\text{input}_{t} = \begin{bmatrix}
 x_{i1}^{\text{SO}_2} & x_{i1}^{\text{NO}_x} & x_{i1}^{\text{PM}_{10}} & x_{i1}^{\text{PM}_{2.5}} & x_{i1}^{\text{CO}} & x_{i1}^{\text{O}_3} \\
 x_{i2}^{\text{SO}_2} & x_{i2}^{\text{NO}_x} & x_{i2}^{\text{PM}_{10}} & x_{i2}^{\text{PM}_{2.5}} & x_{i2}^{\text{CO}} & x_{i2}^{\text{O}_3} \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
 x_{iM}^{\text{SO}_2} & x_{iM}^{\text{NO}_x} & x_{iM}^{\text{PM}_{10}} & x_{iM}^{\text{PM}_{2.5}} & x_{iM}^{\text{CO}} & x_{iM}^{\text{O}_3} \\
 \end{bmatrix}
\]

where \( x_{ij}^{\text{SO}_2}, x_{ij}^{\text{NO}_x}, x_{ij}^{\text{PM}_{10}}, x_{ij}^{\text{PM}_{2.5}}, x_{ij}^{\text{CO}}, x_{ij}^{\text{O}_3} \) is the concentration value of \( \text{CO}, \text{NO}_x, \text{SO}_2, \text{O}_3, \text{PM}_{10}, \text{PM}_{2.5} \) at time \( J \).

(4) The spatial correlation features extracted by KNN \( x_{ij}^{sp} \). As a second input, it is spliced with the output of the Prophet-LSTM.

(5) Let \( K = K + 1 \), repeat steps (3-4) until \( K > M \) sites, where \( M \) is the number of neighboring sites.

(6) Determine the optimal \( K \) according to the evaluation index (selecting the RMSE).

3. Results and discussion

3.1. Monitoring station and data display

The data of this study come from the China Environmental Monitoring Station (http://www.cnemc.cn/), and the daily data of pollutant concentrations in Wuhan city from January 1, 2014, to May 3, 2021, are selected. The total amount of data is 2678, and there are no missing values. The historical data from January 1, 2014, to December 30, 2020, are used as the training set, and the data from January 1, 2021, to May 3, 2021, are used as the test set. A training set is adopted to fit the model parameters. The test set evaluates the predictive ability of the model. Site names and distribution are shown in Table 1 and Fig. 3.

According to the correlation of the \( O_3 \) time series of each station, the correlation coefficient between Wujiaoshan station and Zhuankou station and other stations is not more than 0.6, while the correlation coefficient between any two other stations is more than 0.7. Therefore, the above two stations are ignored in the selection of spatial stations. The data distributions are shown in Fig. 4.

3.2. Selection of evaluation criteria

Mean absolute error (MAE), root mean square error (RMSE) and mean square error (MSE) are selected to measure the prediction accuracy of different models. MSE squares the error (let \( e = \) true value - predicted value), so if \( e > 1 \), MSE will further increase the error. If there are outliers in the data, the value of \( e \) will be large, and \( e \) will be much larger than \(|e|\). Therefore, the model using MSE assigns more weight to outliers than using MAE to calculate the loss. However, this reduces the overall performance of the model. MAE loss works better if the training data are polluted by outliers (e.g., there are many false negative and positive labels in the training data but not in the test set). \( y_i \) is a true value, \( \hat{y}_i \) is an estimate, \( i = 1, 2, \ldots, n \). Among \( n \) is the sample size, and the expression of the above indicator is shown in Eq. (14)-(Eq. (16)).

\[
\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|
\]

\[
\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}
\]

\[
\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2
\]

According to the expression of the above evaluation index, the smaller the values of the three are, the smaller the prediction error of the model is.

3.3. Experimental environment and model parameter settings

The experimental environment and computer configuration are as follows: the programming language is Python 3.8, the development environment is Visual Studio Code editor, and the operating system is Windows 10 x64.

The time step of the LSTM layer is 12, and the \( K \) value is selected as 3. Specific network parameter settings are shown in Table 2.

3.4. Experimental results and analysis

3.4.1. Experimental results

According to Prophet’s decomposition principle, the periodicity is decomposed down. As shown in Fig. 5, from the annual trend, the concentration of \( O_3 \) in Wuhan in winter and spring is higher than that in summer and autumn, with the lowest concentration in July and the highest concentration in February. The rationality and accuracy of the model are further verified by the analysis of the decomposition results.

This experiment has been predicted for nearly three months, and the fitting effect of each model is shown in Fig. 6. The traditional ARIMA model and the linear Prophet model are not sensitive to the extreme

| Station ID          | Classification | Latitude | Longitude |
|---------------------|----------------|----------|-----------|
| Qinghan Gonghua Station | industrial    | 114.36   | 30.62     |
| Wuchang Zhiyang Station | urban         | 114.30   | 30.52     |
| Hanyang Moon Lake Station | suburban     | 114.25   | 30.55     |
| 182 National Avenue Station | urban    | 114.40   | 30.46     |
| Zhuankou New District Station | urban       | 114.16   | 30.48     |
| HanKou Flower Bridge Station | urban      | 114.37   | 30.55     |
| East Lake Pear Garden Station | suburban | 114.30   | 30.59     |
| HanKou River Beach Station | industrial  | 114.28   | 30.61     |
| Wujiaoshan Station       | industrial    | 114.12   | 30.63     |
values of the data, and their fitting curves are relatively smooth. However, the prediction of LSTM is unstable, and its volatility is too large, which leads to poor model fitting. Compared with the single model, the hybrid model Prophet-LSTM. The variance of the prediction results is reduced, and the accuracy is also improved because after the time series data are decomposed, the LSTM only predicts the error term, and the seasonal term and the trend term are linearly fitted by The Prophet, thus significantly improving the prediction effect of the model. The target model KNN-Prophet-LSTM selects the spatial relevant information of the target site through the KNN algorithm and utilizes the characteristic that LSTM has time memory. The prediction accuracy of the

| Layer       | Output shape | Param  |
|-------------|--------------|--------|
| InputLayer  | (12, 6)      | 0      |
| LSTM_output | 64, 18176    |        |
| Aux_input   | 2, 0         |        |
| Concatenate | 66, 0        |        |
| Dense_1     | 6, 402       |        |
| Output      | 1, 7         |        |
model is further improved by considering the two dimensions of time and space.

3.4.2. Model comparison and analysis

In the experiment, we compare the performance of five different models and choose the ARIMA as the baseline model. Among the selected methods, ARIMA is representative of statistical learning methods, and Prophet combines statistical learning methods and deep learning methods. The neural network represented by LSTM is the most used deep learning method for solving time series forecasting problems. Therefore, selecting these methods for comparison can fully verify the effectiveness of the prediction effect of KNN-Prophet-LSTM. In the experiment, the hyperparameters of all the comparison methods are adjusted to the optimal values according to the suggestions of the original paper. The comparison results are shown in Table 3. Under the two evaluation indexes, the target model KNN-Prophet-LSTM shows the best performance, in which the MAE and RMSE are 10. 9009 and 14. 7334, respectively, which are 11. 183 and 12. 8043 lower than the second-best model (the Prophet-LSTM), respectively. Compared with the baseline model ARIMA, the accuracy is improved by approximately 49.76% (MAE) and 46.81% (RMSE), respectively. The reason for the analysis is mainly because the data set selected in this paper comes from air quality monitoring data in the environmental field, and the nature of pollutants often shows strong daily, monthly and annual cycles, which is more in line with the assumption of KNN-Prophet-LSTM. Using the model in this scenario can better capture the periodicity of the time series to obtain a more ideal forecasting effect. It is worth noting that although the LSTM has good accuracy in dealing with various types of time series, its effect is even worse than that of the traditional ARIMA model in predicting air pollutant concentration data. In addition, through further comparison, it is found that the hybrid model has
better performance than the single model, and the improvement of prediction accuracy is more obvious.

4. Conclusion

In this paper, a prediction method based on the KNN-Prophet-LSTM combination model is established by using the pollutant concentration data of Wuhan city and considering the characteristics of time and space. The method comprises the following steps: dividing data into a trend item, a period item and an error item through the Prophet decomposition method, comprehensively considering the advantages of the Prophet model and the LSTM model, predicting the trend item and the period item by using the Prophet model, and predicting the error item by using the LSTM model. The KNN algorithm is added to fuse the spatiotemporal information and predict the $O_3$ concentration value day by day. To highlight the effectiveness and rationality of the KNN-Prophet-LSTM hybrid model, four groups of comparative experiments are set up to compare it with the single model ARIMA, Prophet, LSTM and the hybrid model Prophet-LSTM. The results show that (1) the daily maximum 8-hour average concentration of $O_3$ in Wuhan has a significant periodic change. The difference in the surrounding environment will lead to the difference in $O_3$ concentration change in the region, and the $O_3$ concentration change of similar stations will have a high similarity. (2) The Prophet decomposition algorithm decomposes the original time series, which can effectively extract the time series information and remove the noise so that the prediction accuracy is significantly improved. (3) Through a KNN algorithm, considering the spatial information of the surrounding sites, the accuracy of the model can be further improved. Compared with the baseline model ARIMA, the accuracy is improved by approximately 49.76% (MAE) and 46.81% (RMSE), respectively. (4) The prediction effect of the mixed model is generally better than that of the single model, and the prediction accuracy is higher. Although the KNN-Prophet-LSTM proposed in this paper has high accuracy for the concentration prediction of $O_3$ time series data, it does not extract too much for the error term, resulting in more factors included; thus, the interpretation of the error term is poor. If the error term can be further decomposed, the prediction accuracy may be further improved.

Declarations

Author contributions statement

Biao Zhang: Conceived and designed the experiments. Chao Song: Performed the experiments; Wrote the paper. Ying Li: Analyzed and interpreted the data. Xuchu Jiang: Contributed reagents, materials, analysis tools or data; Wrote the paper.

Funding statement

Biao Zhang was supported by Natural Science Foundation of Shandong Province [ZR2021QFO36], “Guangyue Young Scholar Innovation Team” of Liaocheng University [LCUGYTD2022-03].

---

Data availability statement

Data included in article /supp.material/ referenced in article.

Declaration of interests statement

The authors declare no conflict of interest.

Additional information

No additional information is available for this paper.

References

[1] A. Yafouz, N. AlDahoud, A.H. Birima, et al., Comprehensive comparison of various machine learning algorithms for short-term ozone concentration prediction, Alex. Eng. J. 61 (6) (2022) 4607–4622.

[2] X. Ren, Z. Mi, T. Cai, et al., Flexible Bayesian ensemble machine learning framework for predicting local ozone concentrations, Environ. Sci. Technol. 56 (7) (2022) 3871–3883.

[3] R. Kurniawan, I.N. Setiawan, R.E. Caraka, et al., Using Harris hawk optimization towards support vector regression to ozone prediction, Stoch. Environ. Res. Risk Assess. 36 (2) (2022) 429–449.

[4] D. Li, X. Ren, Prediction of ozone hourly concentrations based on machine learning technology, Sustainability 14 (10) (2022) 5964.

[5] J. Zhou, Z. Xu, S. Wang, A novel dual-scale ensemble learning paradigm with error correction for predicting daily ozone concentration based on multi-decomposition process and intelligent algorithm optimization, and its application in heavily polluted regions of China, Atmos. Pollut. Res. 13 (2) (2022) 101306.

[6] H. Qu, R. Zhang, Short-term mathematical prediction model of air quality based on CEEMD-ELM-PSO, in: 2022 IEEE International Conference on Electrical Engineering, Big Data and Algorithms (EEDA), IEEE, 2022, pp. 227–232.

[7] W. Ma, Z. Yuan, A.K.H. Lai, et al., Optimized neural network for daily-scale ozone prediction based on transfer learning, Sci. Total Environ. 827 (2022) 154279.

[8] W. Mao, L. Jiao, W. Wang, Long time series ozone prediction in China: a novel dynamic spatiotemporal deep learning approach, Build. Environ. 218 (2022) 109087.

[9] M.A. Khan, H. Kim, H. Park, Exploiting neural network for temporal multi-variate air quality and pollutant prediction, J. Korea Multimed. Soc. 25 (2) (2022) 440–449.

[10] P. Muthukumar, K. Nagrecha, E. Cocom, et al., Predicting PM2.5 air pollution using deep learning with multisource satellite and ground-based observations and meteorological and wildfire big data, in: AGU Fall Meeting Abstracts, 2021, GC45B-0842.

[11] Y. Chen, X. Chen, A. Xu, et al., A hybrid CNN-transformer model for ozone concentration prediction, Air Qual. Atmos. Health (2022) 1–14.

[12] D. Ma, S. Ma, Q. Chen, et al., Temperature prediction algorithm based on spatio-temporal prediction, in: 2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design (CSCWD), IEEE, 2022, pp. 151–157.

[13] S. Toskov, M. Lazarova, A. Aleksieva-Petrova, A hybrid spatiotemporal deep model based on CNN and LSTM for air pollution prediction, Sustainability 14 (9) (2022) 5104.

[14] C. Suresh, B.V. Kiranmayee, B. Sneha, Analysis and prediction of air pollutant using machine learning, in: Proceedings of Second International Conference on Advances in Computer Engineering and Communication Systems, Springer, Singapore, 2022, pp. 335–345.

[15] D. Marvin, L. Nespoli, D. Strepparava, et al., A data-driven approach to forecasting ground-level ozone concentration, Int. J. Forecast. 38 (3) (2022) 970–987.

[16] K. Gu, H. Liu, J. Liu, et al., Air pollution prediction in mass rallies with a new temporally-weighted sample-based multitask learner, IEEE Trans. Instrum. Meas. 71 (2022) 1–15.

[17] S.J. Taylor, B. Letham, Forecasting at scale, PeerJ Preprints 5:e3190v2, https://doi.org/10.7287/peerj.preprints.3190v2, 2017.

---

Table 3. Model comparison.

| Model          | MAE     | RMSE   |
|----------------|---------|--------|
| ARIMA (2.0.2)  | 21.7903 | 27.6676|
| LSTM           | 25.8970 | 32.4554|
| Prophet        | 22.9098 | 28.4266|
| Prophet-LSTM   | 21.7179 | 27.5377|
| KNN-Prophet-LSTM | 16.9009 | 14.7334|

---

Additional information

No additional information is available for this paper.