

Air quality forecasting in non-monitored urban areas through machine and deep-learning model

Fernando Illescas-Martinez^{ID}, Laura Garcia^{ID*}, Antonio-Javier Garcia-Sanchez^{ID},
Rafael Asorey-Cacheda^{ID}, Joan Garcia-Haro^{ID}

Department of Information and Communications Technologies, Universidad Politécnica de Cartagena, Plaza del Hospital, 1, Cartagena, 30202, Región de Murcia, Spain

ARTICLE INFO

Keywords:

Air quality
Air pollution forecasting
Deep learning
Machine learning

ABSTRACT

Air pollution poses a major environmental challenge, raising concerns about human health in urban environments. It leads to diseases such as asthma, exacerbates pulmonary conditions, and creates murky skies, lowering inhabitants' quality of life. To quantify air pollution, cost-effective IoT (Internet of Things) devices are being deployed in cities, making air quality monitoring available for a wide range of end-users, including public administrations. However, full urban coverage is unfeasible, and awareness of the carbon footprint of IoT deployments is increasing. Therefore, new techniques are needed to maximize the value of IoT networks with reduced infrastructure. To address these challenges, this paper presents an air pollution analytical forecasting solution based on deep-learning/machine-learning techniques to estimate air quality in locations without deployed devices. Different combinations of well-known deep-learning models are compared with machine-learning techniques to determine the best approach for monitoring polluting gases and airborne particles based on well-defined evaluation metrics. Additionally, two new deep-learning techniques, Multipath-CNN-LSTM (M-CNN-LSTM) and Multipath-CNN-BiLSTM (M-CNN-BiLSTM), are proposed to conduct a more exhaustive comparison. Combinations of LSTM (Long Short-Term Memory) techniques give the best results, with different models working best for each pollutant. Specifically, LSTM was optimal for O_3 , and combinations of CNN (Convolutional Neural Networks) and BiLSTM (Bidirectional LSTM) worked best for NO_2 . GRU (Gated Recurrent Unit) was more efficient for $PM_{2.5}$, and BiLSTM performed best for PM_{10} . This demonstrates that the best strategy to accurately predict the time evolution of each pollutant's behavior depends on the selection of the most suitable machine-learning or deep-learning technique.

1. Introduction

Atmospheric pollution poses a global challenge, significantly impacting major cities and industrial areas where more sources emitting harmful gases and particles are located. These polluting elements are composed of organic and inorganic substances suspended in the air in a combination of liquids and solids (Pražnikar & Pražnikar, 2012). They have adverse effects on human health, vegetation, ecosystems, and the climate, which has led to an increasing interest in deploying air quality monitoring solutions. Air quality monitoring addresses various essential questions aimed at enhancing environmental management and decision-making. It involves measuring air pollution concentrations using specific sensors, revealing the current air quality and establishing historical trends for comparison with national and international air quality standards (García et al., 2022). Furthermore, air quality

monitoring systems provide early alerts by detecting risky situations or surpassed legal/recommended pollution limits, prompting preventive or corrective measures to safeguard public health and the environment. In terms of urban planning, air quality monitoring identifies vulnerable or affected areas and helps devise strategies to curtail emissions, improve mobility, optimize land use, and foster sustainable development. Economically, identifying less polluted areas can boost the growth of a city's business fabric, which could attract environmentally concerned clients. In addition, it contributes to increasing public awareness by informing and educating citizens about air quality-related issues and benefits, promoting responsible and participatory habits and attitudes. Lastly, it provides a tool for verifying compliance with legal and environmental regulations for industrial, commercial, and service activities that generate polluting emissions.

* Corresponding author.

E-mail addresses: fernandille@gmail.com (F. Illescas-Martinez), laura.garcia@upct.es (L. Garcia), antoniojavier.garcia@upct.es (A.-J. Garcia-Sanchez), rafael.asorey@upct.es (R. Asorey-Cacheda), joang.haro@upct.es (J. Garcia-Haro).

<https://doi.org/10.1016/j.eswa.2025.127749>

Received 10 February 2025; Received in revised form 8 April 2025; Accepted 13 April 2025

Available online 29 April 2025

0957-4174/© 2025 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC license (<http://creativecommons.org/licenses/by-nc/4.0/>).

Deploying air quality monitoring systems involves installing meteorological stations in the area of interest. However, meteorological stations with high quality air pollutant sensors are expensive and scarce, which limits the amount of available data. These data are necessary to assess the precise location of the station but do not provide enough information to make decisions addressing specific problems in nearby areas. One way to gather more data from more areas is using cost-effective sensors and IoT (Internet of Things) air quality monitoring solutions, which can be deployed in greater numbers due to their affordability and smaller size. These systems rely on a sensor network strategically positioned at sites representative of population exposure and emission sources (Marinov et al., 2016). The sensors continuously record atmospheric pollutant values and, in some cases, meteorological variables such as temperature, humidity, atmospheric pressure, wind, and solar radiation. Collected sensor data are transmitted to a centralized platform for processing, storage, and real-time information dissemination via media such as websites, mobile apps, and information panels. Nevertheless, growing concerns about the carbon footprint of IoT deployments have led to the search for ways to obtain the most benefit from the smallest possible IoT network.

Artificial intelligence (AI) is revolutionizing the way we approach problems in a wide variety of fields, including air quality monitoring. Machine-learning and deep-learning techniques have been used to analyze air pollution data, firstly to find correlations between pollutants and meteorological factors and, more recently, for forecasting and dealing with incomplete or sparse data (Schürholz et al., 2020). While machine-learning (ML) techniques have better performance when patterns are contrasted with similar objects, deep-learning (DL) techniques are able to find deep and complex relationships among data inputs through efficient training processes (Qin et al., 2019). Therefore, using machine-learning or deep-learning solutions requires accurate problem specification before choosing the most adequate technique. Machine-learning and deep-learning techniques can be powerful on their own, with each technique addressing a particular problem, such as spatial data processing using Convolutional Neural Networks (CNN) or Recurrent Neural Networks (RNN) for sequential data structures (Janniesch et al., 2021). However, the combination of different techniques has proved successful for more complex predictions that consider, for example, data from multiple stations (Lu & Li, 2023) or space-time predictions using images depicting urban air pollution (Le et al., 2020). This has led to interest in comparing models to determine the optimal solution for each pollutant or their combination in the Air Quality Index (AQI), considering both accuracy and computational efficiency (Dong et al., 2024; Wang et al., 2024). However, the current literature has focused on forecasting air quality data in the location of the station (AlShehhi & Welsch, 2023; Bekkar et al., 2021; Dairi et al., 2021). Thus, the challenge of reducing the number of air monitoring devices deployed in cities while determining air pollution levels at any point of the city cost-effectively has remained unaddressed.

To advance in the air quality monitoring field, this work addresses the gap in the literature by proposing a predicting model to estimate air pollution levels at any city location using the data from a small number of IoT devices deployed in other parts of the city. That is, the models presented in this work are used to predict cross-location pollutant concentrations in urban areas where no IoT devices are available, effectively generating air quality values as if from a virtual sensor. This allows users to access reliable data even in the absence of extensive monitoring infrastructure. The aim is to enhance the precision and spatial resolution of pollutant distribution, enabling affected areas to be identified for more effective air quality improvement measures. This contrasts with existing works, which primarily focus on temporal predictions for a limited set of pollutants, AQI forecasting, tracking the spatial and/or temporal evolution of specific pollutants in areas with monitoring stations, or specifically addressing traffic-related emissions (Agbehadjji & Obagbuwa, 2024). The predictive model leverages advanced machine learning and artificial neural network techniques,

and it is capable of learning intricate patterns from large datasets. This study utilizes historical and real-time air quality data from London's monitoring stations, covering particulate matter and gaseous pollutants. Meteorological variables (temperature, humidity, atmospheric pressure, wind, solar radiation) and distance from the target location will be considered, as well as how the evolution of pollution concentrations are influenced by them. The main contributions of this paper can be summarized as follows.

- The dataset has been created from public data of real IoT air quality monitoring devices deployed in the city of London.
- The pollutants have been selected in accordance with the dataset, discarding pollutants with insufficient information.
- A set of machine-learning and deep-learning techniques has been studied to determine the one with the best performance for each pollutant regarding predictions in areas without IoT monitoring coverage.
- Two new space-time deep-learning techniques have been designed and developed that show better performance than existing techniques.

The rest of the paper is organized as follows. Section 3 details the methodology followed to obtain the relevant data, create the proposed models, and train the models. The results for each pollutant are discussed in Section 4, comparing the performance of each model. Conclusions summarizing the main findings are written in Section 5.

2. Related work

Air quality forecasting in urban environments has attracted significant attention due to its profound implications for public health and environmental management. Furthermore, recent advances in Artificial Intelligence (AI) and sensor technology have facilitated the development of innovative solutions. Machine learning, as a subset of AI techniques, has been thoroughly studied to assess its suitability for this purpose. Elena Mitreska et al. examined the measurement and prediction of air pollution levels in urban settings, using a systematic review methodology to analyze the relevant literature (Mitreska Jovanovska et al., 2023). Through a comparison of various ML algorithms and statistical approaches employed in previous studies, the research highlighted reliable methods of air pollution measurement and prediction. The findings underscored the feasibility of leveraging technology to overcome technological hurdles and provide decision-makers with actionable insights to effectively combat air pollution. Lakindu Mampitiya et al. employed machine learning algorithms, including XGBoost, CatBoost, LightBGM, LSTM, and GRU (Gated Recurrent Unit) (Mampitiya et al., 2023), to forecast PM_{10} levels in highly impacted urban areas of Sri Lanka. The study revealed the importance of developing tailored forecasting models for specific geographical regions, with LightBGM demonstrating superior performance in Battaramulla and Kandy (Sri Lanka).

More recent works have focused on the use of deep-learning techniques. Abdelkader Dairi et al. presented a deep-learning model for air pollution forecasting based on Integrated Multiple Directed Attention (IMDA) (Dairi et al., 2021). The authors predicted the presence of four polluting gases (NO_2 , O_3 , SO_2 , and CO) in US states and assessed the performance of their proposal with six statistical indicators. The predictions were compared to other deep-learning techniques such as LSTM, GRU, VAE (Variational AutoEncoder), and a combination of them. IMDA-VAE showed better performance than the other solutions based on unidirectional and bidirectional RNN (Recurrent Neural Networks). A deep-learning based application for air pollution forecasting was developed by Azim Heydari et al. and LSTM and Multi-Verse Optimization Algorithm (MVO) techniques were combined to assess NO_2 and SO_2 air polluting gases from combined cycle power plants, with temperature and wind speed as additional inputs (Heydari et al., 2022). The model

was compared with ENN-PSO (Elman Neural Network-Particle Swarm Optimization), ENN-MVO, and LSTM-PSO models, and it showed the best performance. Jayaraman and Abirami (2025) proposed a Time-Based Spatial (TBS) framework for AQI forecasting using a hybrid CNN and ARIMA (AutoRegressive Integrated Moving Average) model. These techniques were employed to capture spatial dependencies based on latitude and longitude, and temporal dependencies based on pollutant concentrations, respectively. The dataset included various pollutants along with latitude, longitude, date, and time information from 22 different cities across India. The model was evaluated using R-square (R^2), MAE (Mean Absolute Error), RMSE (Root Mean Squared Error) and MAPE (Mean Absolute Percentage Error) metrics. No information on computational costs was provided. The results demonstrated that the proposed framework outperformed other models in terms of MAPE and achieved performance comparable to the Multi-Factor LSTM model in terms of R-square. However, it did not surpass other approaches across the remaining evaluation metrics.

Air pollution data are not the only type of data that can be used to train air quality forecasting models. Several studies have explored using multiple environmental factors and data from urban activities as inputs for their AI models. Saad Hameed et al. proposed a multi-modal framework for urban air quality prediction, integrating real-time data from environmental sensors and traffic density extracted from Closed Circuit Television footage (Hameed et al., 2023). The framework effectively addressed data inconsistencies and sensor failures by employing a Particle Swarm Optimization (PSO)-based merit fusion technique. The study demonstrated notable improvements over traditional models, achieving enhanced prediction accuracy using variants of the LSTM model, such as Bi-directional LSTM, CNN-LSTM, and Convolution LSTM (ConvLSTM). Van-Duc Le et al. devised a spatiotemporal air pollution interpolation and prediction model based on deep-learning that considers innovative input data such as vehicle traffic volume and average driving speeds (Le et al., 2020). The proposed model is a combination of CNN and LSTM that permits temporal and spatial features to be managed. The input data are transformed into a series of images that are later fed into the model. The proposed ConvLSTM model outperformed other models, such as DAL (Deep Air Learning) and CNN encoder-decoder. It obtained the best results using only air pollution and meteorological data, with the option of employing all the available data (air pollution, meteorology, traffic, speed, and external data) as a close second. Saheer Babu et al. proposed a comprehensive framework for air quality modeling, leveraging big data from various domains, including pollution concentrations, urban traffic, aerial imagery, and weather conditions (Babu Saheer et al., 2022). The framework incorporated advanced AI techniques and a novel vegetation detection system using deep learning. The pilot study conducted in the UK city of Cambridge demonstrated the potential of the framework to broaden air quality modeling and inform policies for climate change mitigation and sustainable urban planning.

Most previous studies have used data from one meteorological station to perform predictions in the same location as the station. Conversely, other works use data from multiple stations to forecast air quality at only one selected station. Abdellatif Bekkar et al. implemented a deep-learning solution incorporating CNN-LSTM architecture (Bekkar et al., 2021) to predict $PM_{2.5}$ concentrations in Beijing, China. By integrating the historical data of pollutants, meteorological data, and $PM_{2.5}$ concentrations from adjacent stations, the hybrid CNN-LSTM model outperformed traditional models in predictive performance, emphasizing the efficacy of deep-learning approaches to air quality forecasting. Yanan Lu et al. presented a CNN-BiLSTM model for multistation collaborative air pollution prediction (Lu & Li, 2023). The authors used data from different stations as inputs for predictions in another existing station. SO_2 , NO_2 , O_3 , CO , $PM_{2.5}$, and PM_{10} were considered as input pollutants. The correlation analysis among the stations showed that closer stations presented higher correlations. Single station and multistation CNN-BiLSTM models were compared,

and the results showed that the CNN-BiLSTM model with multistation data had good performance in short-term predictions of pollutant concentrations. However, CO performed better with data from only one station. Jianquan Xie et al. Xie et al. (2025) conducted spatiotemporal air quality predictions for $PM_{2.5}$. The dataset used was collected from 29 monitoring stations across four different cities in China over a three-year period. Initially, Kriging interpolation was applied to generate spatial distributions of pollutant concentrations in the form of images. These images were then fed into the SwinLSTM model to produce the final predictions. The model was evaluated using MAE, RMSE, and R-squared metrics. The results demonstrated that SwinLSTM outperformed ConvLSTM by 5% in terms of R^2 . However, no information regarding computational costs was provided. Cui-lin Wu et al. Wu et al. (2023) proposed Res-GCN-BiLSTM, a hybrid model that combines a residual neural network with a graph convolutional network and BiLSTM to predict regional concentrations of NO_2 and O_3 by leveraging spatiotemporal dependencies. One year of data on various air pollutants was collected from 72 air quality monitoring stations across Shanghai to train the model. Performance was evaluated using MAE, RMSE, and R^2 . The proposed model outperformed seven baseline models in terms of accuracy, achieving improvements of 11% for NO_2 and 17% for O_3 . However, this work lacks information in relation to computational cost. Lastly, Calo et al. (2024) addressed a problem formulation similar to the one presented in this work. They employed a Graph Neural Network (GNN) approach for spatial air quality prediction using message passing. The dataset was obtained from an open-source air quality database for the city of Barcelona, complemented by data on road sections and intersections. Their model was evaluated using MSE and MAPE, assessing performance under varying percentages of missing variables. The results showed that their proposed Mean Aggregation Message Passing (MAMP) method, when combined with GNN, outperformed all other evaluated methods, which did not include any of the approaches considered in this work (see Table 1).

Compared to the existing literature, the work presented in this paper incorporates a wider range of pollutant and meteorological data — such as wind speed — and also accounts for the distance between multiple IoT devices across different areas of London. This information is used to forecast and interpolate air quality, specifying individual pollutants at various locations in the city, even those without direct monitoring. In other words, the proposed method enables cross-location air quality predictions. This approach contrasts with prior studies, which focus on spatiotemporal predictions at existing stations (Jayaraman & Abirami, 2025; Wu et al., 2023; Xie et al., 2025), use inputs from multiple stations but only generate predictions at those same locations (Bekkar et al., 2021; Lu & Li, 2023), or do not evaluate model performance for individual pollutants (Le et al., 2020). One study tackled a similar spatial approach to air quality prediction but proposed a graph-based solution to address the research problem (Calo et al., 2024). As a result, the focus of existing research is largely limited to specific, pre-defined applications, without considering cross-location predictions — an essential feature for cities with fewer resources to deploy extensive monitoring networks — or addressing these limitations by exploring an alternative approach to that presented in this work. In addition, we introduce two new techniques — M-CNN-LSTM and M-CNN-BiLSTM — aimed at enhancing predictive performance specifically in line with the objectives set forth in this work.

3. Methodology

The aim of this work is to predict air quality at any location, in real time, using the data provided by various monitoring IoT devices distributed throughout the city. This section describes the methodology followed to achieve this objective.

The flowchart of the experimental process is illustrated in Fig. 1. The process begins with data collection from IoT sensor devices, which measure various environmental factors. The collected data include

Table 1
Comparison of proposed approach with existing air pollutant prediction methods considering multiple monitoring stations.

Reference	Research focus	Best model	Performance metrics	Pollutants
Xie et al. (2025)	Spatiotemporal prediction	SwingLSTM	MAE, RMSE & R ²	PM _{2.5}
Jayaraman and Abirami (2025)	Time-based-spatial forecasting	HybridCNN & ARIMA	R ² , MAE, RMSE & MAPE	AQI
Wu et al. (2023)	Regional pollutant predictions based on spatiotemporal dependencies	Res-GCN-BiLSTM	MAE, RMSE & R ²	NO ₂ , O ₃
Bekkar et al. (2021)	Spatiotemporal prediction	CNN-LSTM	MAE, RMSE & R ²	PM _{2.5}
Lu and Li (2023)	Multistation collaborative prediction	CNN-BiLSTM	MAE, RMSE & R ²	PM _{2.5} , PM ₁₀ , CO, (O ₃), NO ₂ , SO ₂
Calo et al. (2024)	Spatial predictions	MAMP+GNN	MSE & MAPE	PM _{2.5}
Our work	Cross-location prediction of pollutant concentrations	GRU (PM _{2.5}) BiLSTM (PM ₁₀) M-CNN-BiLSTM (NO ₂) LSTM (O ₃)	RMSE, MB, R ² , MAE & Accuracy	PM _{2.5} , PM ₁₀ , NO ₂ , O ₃

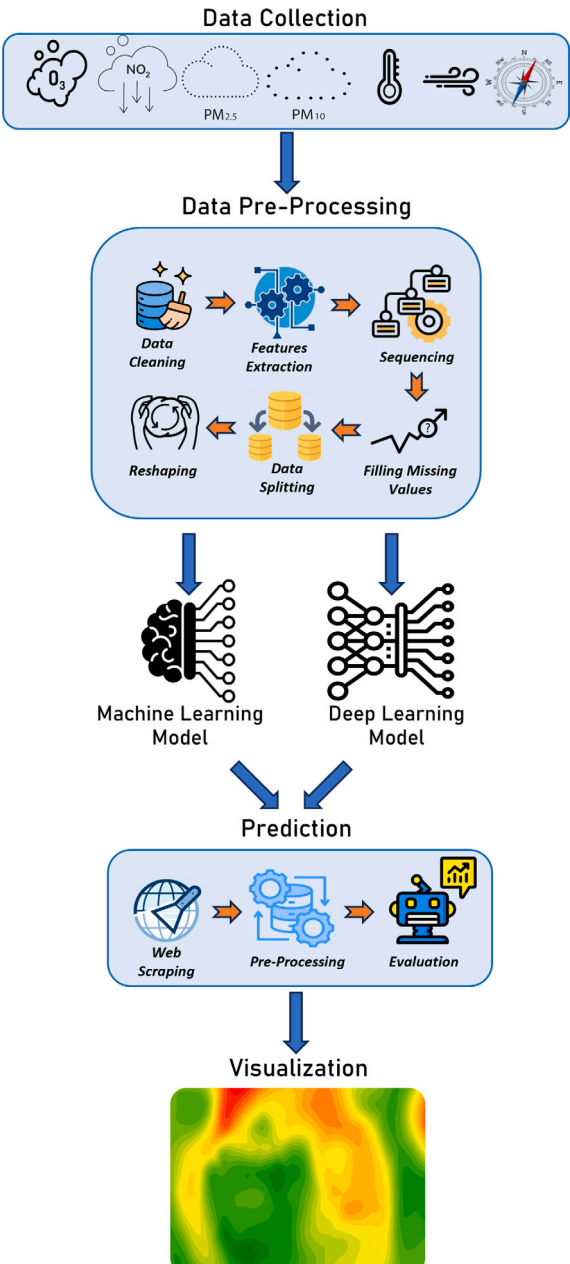


Fig. 1. Flowchart of the experimental process.

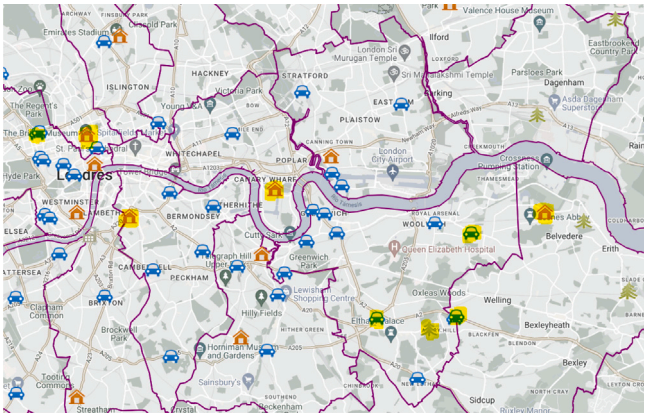


Fig. 2. Study area with the ozone prediction sensor base highlighted.

pollutants like O₃ and NO₂, and suspended particulate matter such as PM_{2.5} and PM₁₀. Data on wind speed, and direction and other meteorological data are also gathered. Once the data are collected, they undergo several pre-processing steps to ensure their quality and usability for analysis. This pre-processing includes data cleaning to remove noise and inconsistencies, feature extraction to identify and select relevant attributes, and sequencing to organize the data in a chronological order. Any missing values in the dataset are filled in, and the data are split into training and testing sets. Finally, the data are reshaped to fit the input requirements of the machine-learning and deep-learning models. After pre-processing, the refined data are fed into various machine-learning and deep-learning models. These models are trained to learn patterns and make predictions based on the input data. The trained models are then deployed to work with the most recent data, gathered through web scraping to automate access to the public data source, providing real-time predictions and insights. The final step involves visualizing the results. The predictions and analyses are displayed in a user-friendly format, enabling easy interpretation and decision-making based on the most up-to-date information.

The action area used in this work is shown in Fig. 2. Symbols, such as cars, houses, and trees, represent different categories of IoT sensor devices, although they are all treated equally. The data they gather include information on pollution and meteorological factors, as required by the pollutant model under study. It is important to note that the data highlighted in yellow are those that will serve as the basis for the prediction, while the remaining data will be used to train the model.

An observation example is illustrated in Table 2. It showcases various elements influencing the prediction: ozone (O₃), nitrogen dioxide (NO₂), particles smaller than 2.5 micrograms per cubic meter (PM_{2.5}), wind speed (WSPD), and the distance from the point being evaluated (Dist) using coordinates. Additionally, for informational purposes, the

Table 2
Sample from the training dataset corresponding to the ozone prediction case.

ReadingDateTime	Site	O ₃	NO ₂	PM _{2.5}	WSPD	Dist	TargetSite
2016-06-30 10:00:00	B - BW	38.3	23.2	7.4	0.3	15.83	H - OS
	C - B	39.8	15.8	6.7	0.3	2.88	
	GW - E	41.1	10.9	6.3	0.3	13.58	
	GW - F	30.1	36.7	7.4	0.3	14.18	
	GW - PHS	38.1	21.5	8.0	0.3	13.25	
	GW - WA	48.3	13.6	8.7	0.4	11.7	
	S - EC	21.4	17.9	7.4	0.3	3.88	
	TH - B	28.5	52.2	7.4	0.2	5.46	
	W - MR	4.1	173.3	7.4	0.3	4.86	

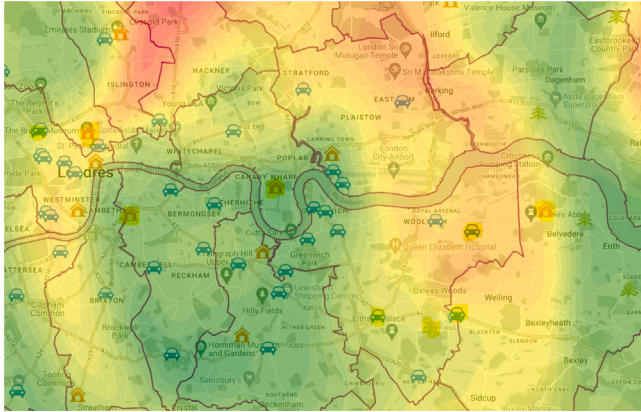


Fig. 3. Study area with an overlaid contour map illustrating the estimated prediction for the area.

date and time (ReadingDateTime), the abbreviated names of the IoT devices used for the prediction (Site), and the target station (TargetSite), which serve as the label, are also shown.

By evaluating the data provided by these highlighted IoT sensor devices, the model is able to estimate pollution levels at different locations. We prefer to visualize this prediction through a heat map, as depicted in Fig. 3.

3.1. Data description

The data collected on air quality for various pollutants have been gathered through the official London Air Quality portal (Environmental Research Group, Imperial College London, 2025). This compiled database includes hourly measurements for each IoT device. Each pollutant has been treated independently as they require different prediction methods. To facilitate predictions, a set of base IoT sensor devices has been selected, providing the necessary data for each pollutant. These sensor devices were chosen based on three criteria: the temporal availability of the data, ensuring the highest number of possible records; a sufficient number of samples for optimal model training; and the spatial distribution of sensors to cover the broadest area of the city. Additionally, multiple models have been selected and two new models have been specifically developed to identify the most suitable predictors for each pollutant, considering the variables accessible to each sensor device.

To facilitate predictions for NO₂, a base set of 12 IoT monitoring devices with the greatest data availability has been employed. These stations were used to estimate NO₂ levels in 65 other sensor devices, acting as labels. The total number of logs is 1,399,656. The predictions are not only performed using NO₂ data, the distance between sensor devices has also been included as a predictor.

For O₃, a base set of 9 sensor devices with the greatest data availability has been selected. These IoT devices were used to calculate O₃ levels in 8 other devices, serving as labels. The total number of logs

is 152,488. The predictors include O₃, NO₂, PM_{2.5}, wind speed, and the distance between stations.

Regarding PM_{2.5}, a base set of 8 IoT devices with the greatest data availability has been chosen. These sensor devices were employed to predict PM_{2.5} levels in 29 other devices, used as labels. The total number of logs is 183,010. The selected predictors include PM_{2.5}, NO₂, O₃, PM₁₀, and the distance between sensors.

Lastly, for PM₁₀, a base set of 15 sensor devices with the greatest data availability has been selected. These IoT devices were used to determine PM₁₀ levels in 51 other sensor devices, employed as labels. The total number of logs is 273,171. The chosen predictors are PM₁₀, NO₂, PM_{2.5}, and the distance between sensors.

To mitigate potential biases caused by missing data or sensor inaccuracies, missing values in variables related to air pollutants were removed from the analysis. Since the volume of available data was sufficient, discarding these records did not compromise the robustness of the model. This decision was based on the assumption that the absence of a specific pollutant might be due to sensor failures, which could introduce noise into the model if imputation were attempted. Specifically, different approaches have been taken for each pollutant. In the case of NO₂, it was not necessary to implement any special treatment since there were a sufficient number of inputs, and in instances where the prediction for a specific hour was unavailable, it was simply omitted. On the other hand, to handle missing values in the PM_{2.5}, PM₁₀, and other meteorological variables, different imputation strategies were explored and their impact on model accuracy was evaluated. Methods such as moving average imputation, temporal interpolation, and machine learning-based algorithms were tested, in addition to the techniques ultimately selected. For PM_{2.5}, it was more appropriate to substitute missing values with a global constant applied across the entire dataset. As for PM₁₀, the choice was to replace the missing values with means computed from the nearest neighbors. The final selection of these methods was based on a comparative analysis of model performance under different imputation strategies, keeping hyperparameters and model architecture constant to ensure a fair evaluation. The results showed that imputation with a global constant for PM_{2.5} resulted in more accurate predictions, possibly due to the lower variability in the missing data patterns for this variable. Conversely, for PM₁₀, imputation using the mean of the nearest neighbors provided better results, suggesting that missing values for this variable were better represented by the local trends in the dataset. Since the goal of preprocessing is to minimize the impact of missing data without introducing unnecessary biases into the model, the final decision was made based on the empirical performance observed in the validation phase, prioritizing the strategy that maximized prediction accuracy. It is crucial to emphasize that this treatment was exclusively applied to the companion predictor variables and not to the main pollutant being forecasted.

Furthermore, during the preprocessing phase, normalization and standardization techniques were explored to assess their impact on model convergence and performance. Approaches such as min-max normalization and standardization based on mean and standard deviation were tested. However, the results showed that applying these techniques did not improve the model's performance and, in some

Table 3
Pairwise overlaps between pollutants.

	NO ₂	PM ₁₀	PM _{2.5}	PM ₁	O ₃	SO ₂
NO ₂	14,531,781 (100.0%)	10,051,229 (69.2%)	3,815,590 (26.3%)	570,960 (3.9%)	3,742,285 (25.8%)	1,456,973 (10.0%)
PM ₁₀	10,051,229 (82.3%)	12,219,599 (100.0%)	3,837,577 (31.4%)	904,218 (7.4%)	3,139,139 (25.7%)	924,623 (7.6%)
PM _{2.5}	3,815,590 (73.7%)	3,837,577 (74.1%)	5,179,703 (100.0%)	777,858 (15.0%)	1,516,453 (29.3%)	302,598 (5.8%)
PM ₁	570,960 (60.2%)	904,218 (95.3%)	777,858 (82.0%)	948,934 (100.0%)	287,395 (30.3%)	0 (0.0%)
O ₃	3,742,285 (95.6%)	3,139,139 (80.2%)	1,516,453 (38.7%)	287,395 (7.3%)	3,913,530 (100.0%)	818,525 (20.9%)
SO ₂	1,456,973 (94.0%)	924,623 (59.7%)	302,598 (19.5%)	0 (0.0%)	818,525 (52.8%)	1,549,246 (100.0%)

cases, led to a significant decrease in prediction accuracy. A possible explanation for this behavior is that the models used in this study are capable of properly handling the original scales of the variables without requiring prior transformation. Additionally, since the input data already exhibit meaningful relationships between the variables, normalization or standardization may have disrupted these relationships in a way that was not beneficial for model learning. Therefore, after an empirical evaluation of different scaling strategies, the decision was made to use the data in its original scale, as this led to better results in terms of model accuracy and stability.

3.2. Statistical study

This subsection presents an exploratory statistical analysis of the variables included in the dataset, with the aim of understanding the underlying structure of the information, identifying potential redundancies, evaluating the completeness of the measurements, and anticipating limitations that may affect the subsequent modeling phase. The approach integrates both an analysis of data availability (matches) and the assessment of linear correlations between variables, all within the context of a real-world dataset, and therefore subject to gaps, measurement errors, and asymmetries inherent to environmental monitoring systems.

Sample overlap between pollutant

A key aspect to consider is the number of shared records between pairs of pollutants. This type of analysis helps identify variable combinations that can be used jointly in models, as well as those whose use may be limited due to a lack of overlapping cases. The analysis showed that NO₂ has a strong data foundation, with over 14 million records, representing the highest coverage in the dataset. PM₁₀ also shows considerable coverage (12 million records), while PM_{2.5} and, especially, PM₁ are significantly more limited in terms of the number of observations.

Regarding pairwise overlaps, NO₂ and PM₁₀ share around 69% of NO₂'s total observations, which allows for their joint use with a reasonable data volume. However, the overlap between NO₂ and PM_{2.5} drops to 26%, and falls sharply with PM₁, with only 3.9%. A similar pattern is seen between PM₁₀ and PM₁ (7.4% overlap), indicating that including PM₁ severely restricts the usable dataset. O₃, with about 3.9 million observations, shows acceptable overlap with NO₂ (25.8%) and PM₁₀ (25.7%). Lastly, SO₂ — although with lower overall coverage — shows moderate overlap with NO₂ (10%) and PM₁₀ (7.6%).

These results highlight that not all variables are equally represented, and that the number of overlapping records can critically impact the type of analysis or modeling that can be performed. As a first approximation, we propose considering combinations with at least 10% overlap relative to the total number of observations for the reference pollutant (see Table 3).

Overlap between pollutants and meteorological variable

The analysis was extended to examine the overlap between pollutants and meteorological variables. The results were more restrictive than in the previous case. In general, most meteorological variables are recorded in only a very small subset of the total observations. For instance, relative humidity (RHUM) and solar radiation (SOLR) show only about 0%–1% overlap with any pollutant, which rules out their direct inclusion in general models without intensive imputation procedures.

On the other hand, variables such as wind direction (WDIR) and wind speed (WSPD) show higher levels of overlap—above 15%, with pollutants like NO₂, O₃, or PM₁₀. In particular, WDIR coincides with O₃ in more than 36% of the records, a figure that, while not ideal, supports its consideration. Barometric pressure (BP) shows some degree of coverage, though limited (below 10% in most cases), while precipitation (RAIN) also appears infrequently, which could restrict its inclusion (see Table 4).

This pattern supports the notion that many meteorological variables cannot be directly used in predictive models without additional preprocessing efforts, such as value imputation, segmentation into complete subsets, or pollutant-specific strategies.

Correlation analysis

As a complement to the coincidence analysis, linear correlations among all variables were explored (see Fig. 4). The correlations between pollutants show patterns consistent with findings in the literature. For example, PM_{2.5} and PM₁ exhibit an extremely high correlation ($r = 0.99$), suggesting that both variables convey virtually the same information. A similar situation is observed between PM_{2.5} and PM₁₀ ($r = 0.87$), which could lead to redundancy in the models if not properly addressed through feature selection or regularization techniques.

The correlation between NO₂ and O₃ is negative ($r = -0.62$), consistent with the photochemical mechanisms of ozone formation, in which NO₂ acts both as a precursor and a sink under low-irradiance conditions. This relationship supports the joint modeling of both pollutants, particularly when temporal or radiative data are available to better capture their interaction.

Regarding the relationships between pollutants and meteorological variables, the coefficients are generally lower, although some associations are noteworthy:

- WSPD shows a positive correlation with O₃ ($r = 0.39$), which may reflect a dispersion effect in open areas.
- RHUM shows a negative correlation with O₃ ($r = -0.67$), supporting the hypothesis that ozone is influenced by humidity, either directly or through its effect on available radiation.
- In contrast, variables such as RAIN or BP do not show significant correlations with the analyzed pollutants, consistent with their limited presence in the dataset.

Table 4
Overlap between pollutants and meteorological factors.

	BP	RAIN	RHUM	SOLR	WDIR	WSPD
NO ₂	731,813 (5.0%)	439,580 (3.0%)	262 (0.0%)	184,774 (1.3%)	2,240,195 (15.4%)	1,454,024 (10.0%)
PM ₁₀	527,523 (4.3%)	260,953 (2.1%)	68 (0.0%)	0 (0.0%)	2,093,963 (17.1%)	1,367,204 (11.2%)
PM _{2.5}	247,108 (4.8%)	14,118 (0.3%)	0 (0.0%)	0 (0.0%)	741,845 (14.3%)	575,701 (11.1%)
PM ₁	0 (0.0%)	14,118 (1.5%)	0 (0.0%)	0 (0.0%)	22,979 (2.4%)	22,979 (2.4%)
O ₃	267,806 (6.8%)	253,005 (6.5%)	60 (0.0%)	0 (0.0%)	1,411,104 (36.1%)	675,708 (17.3%)
SO ₂	219,089 (14.1%)	217,699 (14.1%)	191 (0.0%)	215,732 (13.9%)	234,997 (15.2%)	0 (0.0%)

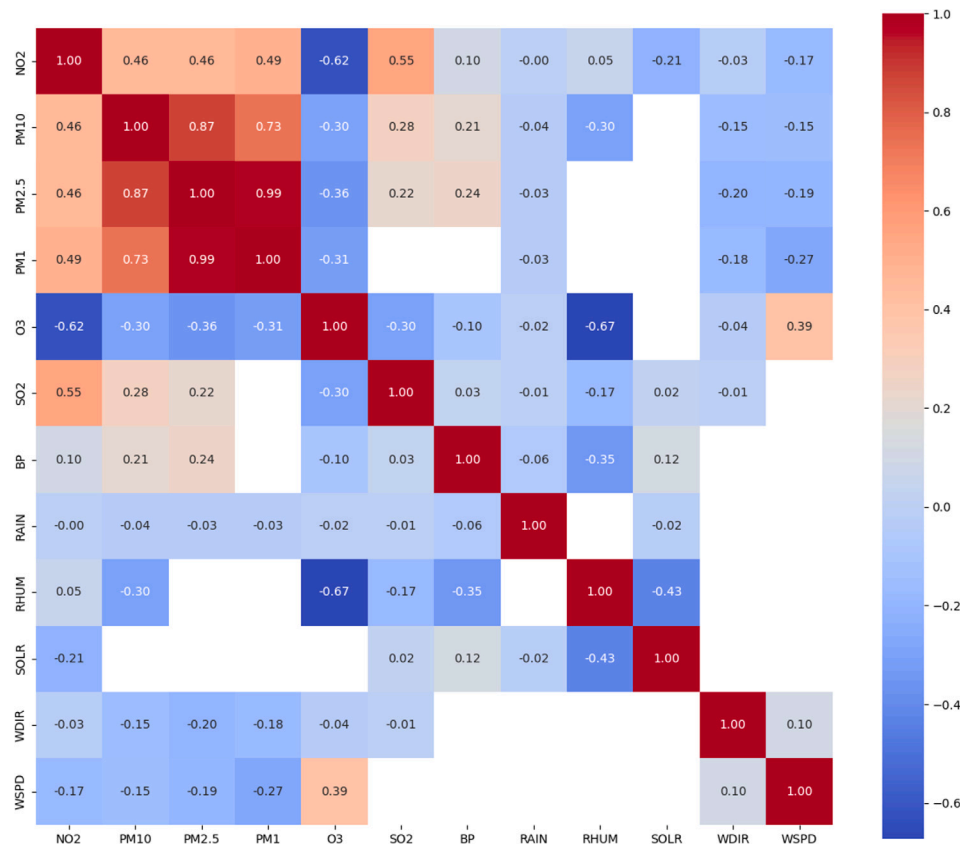


Fig. 4. Heatmap of correlations among all variables.

Modeling considerations

Based on these results, we propose to include a set of variables in the modeling experiments, consisting of NO₂, PM₁₀, PM_{2.5}, O₃, SO₂, WDIR, and WSPD. These variables strike a reasonable balance between the number of observations, cross-matching, and potential correlation with the phenomenon being modeled. However, an iterative and flexible approach will be maintained: the final inclusion of each variable will depend on the performance observed in the models. Additionally, potential imputation techniques for meteorological variables with predictive potential, particularly WDIR and WSPD, will be considered in case there is a need to expand the dataset coverage.

3.3. Descriptive analysis of variable distribution

Despite the usefulness of correlation values, it is also crucial to analyze the individual distribution of the data. To do this, a statistical summary of the key variables was calculated, including mean, median,

standard deviation, quartiles, and extreme values. Below is the table with the descriptive values for each variable (see Table 5).

The analysis reveals a marked heterogeneity in the distributions, both in terms of symmetry and the presence of outliers. The NO₂ variable stands out for its right-skewed distribution. Although the mean is around 40.8 µg/m³ and the median is 32.8 µg/m³, suggesting a relatively moderate central tendency, extreme values exceeding 1200 µg/m³ are observed. This behavior reflects the presence of occasional episodes of high concentration that could significantly impact the models if not properly handled.

Similarly, the suspended particles PM₁₀ and PM_{2.5} exhibit distributions with long tails. For PM₁₀, the median is 17.0 µg/m³, with a maximum value exceeding 3000 µg/m³, while PM_{2.5} shows a median of 9.0 µg/m³ and maximum values exceeding 600 µg/m³. These extreme values, more noticeable in the statistics than in the central mass of the data, reinforce the idea of strongly skewed distributions influenced by occasional events.

Table 5
Statistical summary of variables.

Variable	N	μ	Me	σ	Min	P ₂₅	P ₅₀	P ₇₅	Max
NO ₂	14,542,549	40.78	32.8	32.69	-73.5	17.7	32.8	54.8	1217.3
PM ₁₀	12,259,789	20.68	17.0	15.02	-46.9	11.5	17.0	25.8	3292.8
PM _{2.5}	5,215,538	11.82	9.0	10.15	-99.0	5.7	9.0	14.6	685.0
PM ₁	948,976	6.55	4.0	7.73	-3.0	2.3	4.0	7.6	252.2
O ₃	3,951,554	35.20	34.0	23.56	-41.1	16.0	34.0	51.6	259.0
SO ₂	1,715,564	2.99	2.6	4.47	-17.6	0.9	2.6	4.7	638.3
BP	804,133	1010.18	1011.0	10.85	957.0	1004.0	1011.0	1018.0	1043.0
RAIN	512,237	0.10	0.0	1.08	0.0	0.0	0.0	0.0	123.0
RHUM	336	24.16	26.0	29.70	-103.0	4.0	26.0	45.0	76.0
SOLR	237,974	47.68	9.0	97.51	-101.0	5.0	9.0	46.0	1039.0
WDIR	2,543,935	195.79	209.0	95.11	0.0	115.0	209.0	267.0	360.0
WSPD	1,693,894	1.10	0.8	1.15	0.0	0.3	0.8	1.6	232.0

In contrast, ozone (O₃) presents a more symmetric distribution, with the mean and median being very close (35.2 $\mu\text{g}/\text{m}^3$ and 34.0 $\mu\text{g}/\text{m}^3$, respectively), suggesting a more stable temporal behavior. However, outliers are also recorded, though less pronounced than in the case of the particles.

Sulfur dioxide (SO₂) combines a low mean (2.99 $\mu\text{g}/\text{m}^3$) with a relatively high standard deviation (4.47 $\mu\text{g}/\text{m}^3$) and a maximum value exceeding 600 $\mu\text{g}/\text{m}^3$. This high dispersion points to the influence of highly localized sources or occasional industrial emissions.

As for the meteorological variables, wind direction (WDIR) covers almost the entire scale from 0° to 360°, with a median at 209° and a high standard deviation (95.1°), as expected for a circular variable. Its distribution suggests dominant circulation patterns, though with significant variability. On the other hand, wind speed (WSPD) shows a mean of 1.1 m/s and values centered around 0.8 m/s, but with unrealistic maximum values (up to 232 m/s), which can be attributed to sensor errors.

3.4. Architectures of the proposed multipath neural network models

Machine learning algorithms are mathematical techniques that offer several approaches to handle complex datasets and uncover meaningful patterns, performing tasks such as classification, regression, clustering, and generation. This work has assessed the performance of five commonly used machine learning algorithms across diverse fields to determine air quality in unmonitored urban areas, namely, Decision Tree (DT) (Kingsford & Salzberg, 2008), Random Forest (RF) (Rigatti, 2017), Gradient Boosting (GB) (Natekin & Knoll, 2013), Support Vector Regression (SVR) (Awad et al., 2015), and Multilayer Perceptron (MLP) (Popescu et al., 2009).

Furthermore, strategies incorporating bidirectional layers (Schuster & Paliwal, 1997) enable neural networks to process data not only in a forward manner but also in reverse, capturing broader contexts and enhancing their ability to comprehend intricate relationships within datasets. The following models from the literature have been evaluated in this work for air quality prediction and extrapolation in different city locations: CNN (Gu et al., 2018), GRU (Cho et al., 2014), GRUx2 (Estalayo et al., 2019), LSTM (Hochreiter & Schmidhuber, 1997), LSTMx2 (Zaremba et al., 2014), LSTMGRU (Aslam et al., 2022), and BiLSTM (Siarni-Namini et al., 2019).

CNNs are particularly effective for learning spatial features and local patterns within structured data. In the context of air quality forecasting, CNNs help extract relevant spatial correlations from input features such as pollutant levels and meteorological variables measured across different sensors or locations. These local dependencies capture complex, non-linear relationships that other models may overlook. Moreover, while LSTM networks were originally designed for temporal sequences, they can also be adapted to model spatial sequences or dependencies across measurement points when the data is presented in a structured order. This allows the network to learn how variations in pollutant levels at one location may influence or relate to those in neighboring areas.

Moreover, unlike CNNs, where a filter is used to update the weights, GRU and LSTM are recurrent networks that use gates (internal mechanisms that regulate the flow of information inside the network) to modify the hidden state and determine to which extent the previous hidden state is considered, or even forgotten (LSTM). The main difference between GRU and LSTM is that LSTM has three gates (input, output, and forget), while GRU has only two (update and reset). This makes GRU simpler and faster than LSTM, but also less precise and flexible. The choice in recurrent layers, specifically between LSTM and GRU, depends on the complexity and characteristics of the data and the task. According to some studies (Zhou et al., 2016), GRU tends to perform better on low-complexity sequences, while LSTM excels on high-complexity sequences that require deep context understanding. However, there is no definitive answer, and the best way to find out which one is more suitable is to perform a hyperparameter search. This is a process where a set of different values from the possible external configuration options for training a neural network are tested until reaching the best combination. In general, LSTM offers more control over the network, but also more challenges in terms of training and optimization.

Therefore, in an effort to achieve greater accuracy predicting some of the pollutants assessed in this work, two new models combining CNN and LSTM are introduced. Both models separate the inputs into two paths with different layer selection, the results of which are later concatenated. The synergy of these diverse bidirectional layers, configured in various combinations, constitutes the core of the methodological approach followed in this work.

3.4.1. Multipath CNN-LSTM

The M-CNN-LSTM model has two processes or paths that process the input through a different set of layers and later concatenate the outputs from these paths for further processing (see Fig. 5). The first path begins with a one-dimensional convolutional layer to extract features from the input time series, followed by a max pooling layer to reduce dimensionality while maintaining the most relevant features. After that, a flatten layer transforms the output of the previous layer into a one-dimensional vector. At the same time, the second path applies an LSTM layer to capture temporal dependencies in the data, along with a dropout layer to prevent overfitting. Subsequently, another dense layer is added to obtain a more compact representation of the data. Finally, the output of the path with the LSTM layer is concatenated with the output of the path with the convolutional layer, and it is passed through a fully connected layer with ReLU as the activation function to integrate the features and obtain the final output prediction. The model is compiled with the RMSprop optimizer to minimize the MSE loss function that assesses the error between predictions and the real outputs.

As later discussed, the results from this model are equivalent to other traditional models. The insight gained by developing this model was used to create an evolved version of it, maintaining the two paths that process the inputs while following an updated combination of layers to increase the resulting accuracy.

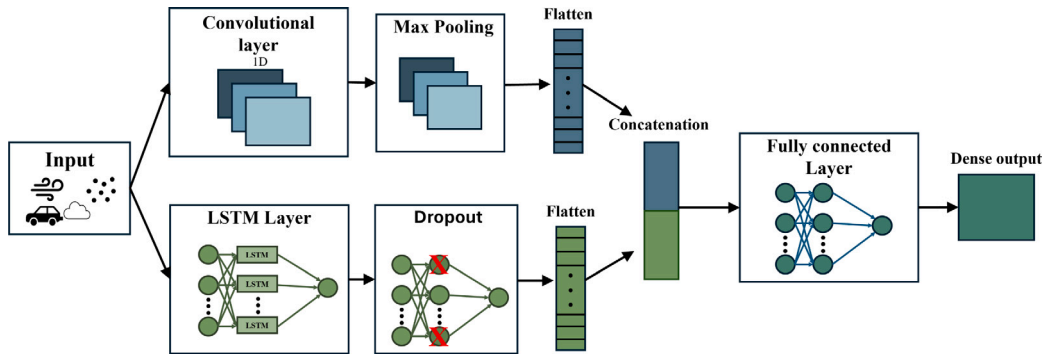


Fig. 5. Architecture of the M-CNN-LSTM model.

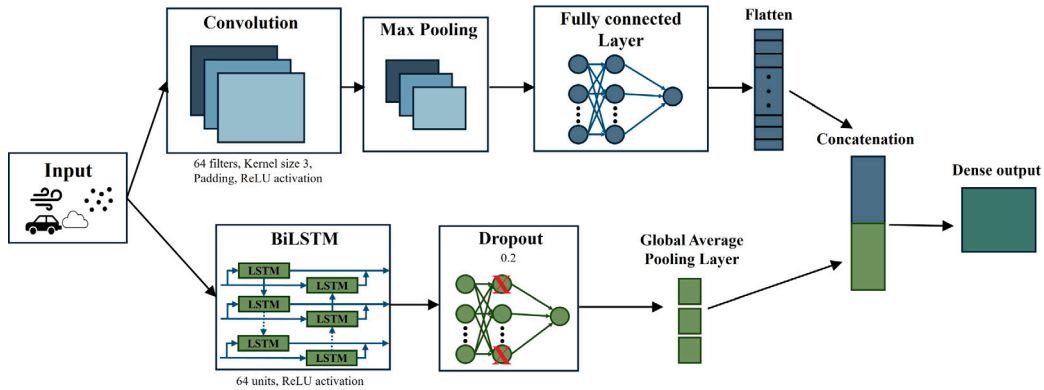


Fig. 6. Architecture of the M-CNN-BiLSTM model.

3.4.2. Multipath CNN-BiLSTM

This model employs a combination of convolutional, recurrent, and concatenation layers to process a time series with multiple features, following the same concept of processing the input through two paths as presented in the previous model (see Fig. 6). The input is a tensor that represents feature values at each time step. The first path begins with a convolutional layer that applies 64 filters (size 3) with ReLU for the activation function and padding to maintain the same dimension. A max pooling layer is then applied to reduce the dimension by half. Subsequently, a dense layer with 32 units is added, and the result is flattened. The second part of the model consists of a bidirectional LSTM layer with 64 units, with ReLU as the activation function, which processes the input recurrently in two directions, from the beginning to the end of the input and from the end to the beginning. A dropout layer with a rate of 20% is applied to prevent overfitting. Next, a global average pooling layer is applied to obtain an output vector. Finally, the results from the convolutional and recurrent paths are concatenated and passed through a dense layer with a single unit and ReLU activation to obtain the final value. The model is compiled with the RMSprop optimizer and MSE loss function.

This model differs from the M-CNN-LSTM architecture due to several key enhancements. Firstly, we use Bidirectional LSTM (BiLSTM) instead of LSTM, enabling the model to capture dependencies from past and future contexts for a more comprehensive understanding of the sequence data. Secondly, we add a dense layer within the convolutional process to improve learning capacity and capture more intricate patterns. Additionally, we reposition the dense layer, moving it to the end of the convolutional process before concatenation with the recurrent process outputs, allowing for a more focused transformation of the convolutional features. Lastly, we replace the flatten layer in the recurrent process with a Global Average Pooling (GAP) layer. This change reduces the dimensionality while preserving important features, aligning the parameter numbers between the convolutional and recurrent processes for better integration and efficiency.

Furthermore, with regard to the hyperparameter tuning of both models, it was performed using a Random Search strategy due to the large number of models evaluated and the computational complexity that would have arisen from an exhaustive search using Grid Search. This approach allowed for the exploration of a wide range of configurations without incurring excessive computational costs, resulting in an efficient selection of optimal values for each model. During the tuning process, the effects of various key hyperparameters were analyzed, including the number of hidden units in LSTM/GRU, filter sizes in CNN, dropout rates, and learning rates. It was observed that an excessive increase in the number of hidden units tended to lead to overfitting, while an excessive reduction impaired the model's ability to capture complex patterns. For this reason, the final selection of hyperparameters was based on a balance between model accuracy, training time, and robustness against overfitting. Additionally, preliminary tests were conducted to evaluate the influence of different hyperparameters on model performance. The final combination was chosen based on the performance observed in cross-validation, prioritizing configurations that yielded good results without compromising the model's generalization. Therefore, although exhaustive optimization was not performed, the strategy employed allowed for effective exploration of the hyperparameter space, ensuring a good trade-off between accuracy and computational efficiency.

4. Experimental process and results

This section presents the experimental setup and the respective results for each pollutant.

4.1. Experimental setup

The experimental setup was carried out on a server with extensive computing resources, featuring a Linux operating system. The

hardware configuration included 256 GB RAM and a 4TB hard drive, driven by four Intel Xeon Gold 5220 2.2G 18C/36T processors with a speed of 10.4GT/s and a 24.75M Cache. The system also integrated 8 TESLA M10 graphic processing units (GPUs) to enhance computational capabilities. All models were trained with the same computational resources.

The software environment was managed within a Conda setup, designed to support Python 3.11 and essential libraries for effective data handling and artificial intelligence model training. This streamlined setup, coupled with robust hardware, served as the foundation for executing and analyzing the experiments in this study.

4.2. Evaluation metrics

Assessing the performance of predictive models is carried out using a diverse set of metrics that compare predicted values with actual observations. These metrics serve as benchmarks to evaluate the accuracy, reliability, and overall effectiveness of various machine-learning techniques when predicting air pollution levels. The selection of these particular metrics was based on those most commonly used in the related literature. The metrics used are the following.

4.2.1. Root Mean Squared Error (RMSE)

It calculates the average deviation between predicted and actual values. It typically ranges from zero to higher values, with an optimal result for lower RMSE values indicating better model performance.

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

4.2.2. Mean Bias (MB)

It measures the average difference between predicted and observed values. Ideally, the MB should converge close to zero.

$$MB = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i) \quad (2)$$

4.2.3. Coefficient of determination (R^2)

It evaluates the proportion of variance captured by the model, with higher values (closer to 1) indicating better model fit (James et al., 2013).

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (3)$$

4.2.4. Mean Absolute Error (MAE)

It computes the average absolute differences between predicted and actual values, aiming for lower values indicating enhanced model accuracy.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (4)$$

4.2.5. Accuracy

It considers the air quality levels associated with each pollutant. This metric gauges the model's effectiveness in accurately assessing the AQI reflected within the evaluated area. The AQI thresholds associated with each pollutant are detailed in Table 6 (Ayuntamiento de Madrid, 2023). This precision metric defines a prediction as accurate when it falls within the range of the actual hazard level. Therefore, the closer the precision value to 1, the more accurate the model in reflecting actual hazard levels.

$$Accuracy = \frac{1}{n} \sum_{i=1}^n 1(\hat{y}_i \in a_i), \quad (5)$$

where function $1(x)$ is the indicator function that takes the value of 1 if x is true and 0 otherwise.

Table 6

Air quality index of different pollutants.

Pollutant	Very good	Good	Average	Bad	Very bad
PM _{2.5}	0–15	16–30	31–55	56–110	> 110
PM ₁₀	0–25	26–50	51–90	91–180	> 180
NO ₂	0–50	51–100	101–200	201–400	> 400
O ₃	0–60	61–120	121–180	181–240	> 240

In these metrics assessing predictive model performance, several variables are consistently employed. y_i represents the observed or actual values of the parameter being predicted, while $\hat{y}_i$ signifies the predicted values generated by the model. The variable n denotes the total number of observations or data points in the dataset under analysis. $\bar{y}_i$ refers to the mean or average value of the observed dataset, and a_i is the actual range of hazard level values for the i th sample.

Support is also drawn from Table 7 provided by the CALIOPE system (Arévalo, 2023), which offers different quality criteria based on specific statistical parameters to evaluate the defined systems, considering previous studies and established results.

Finally, the training time ($TIME_{T_r}$), expressed in seconds, is incorporated to measure the model's training speed. Additionally, the testing time ($TIME_{T_e}$) is assessed, providing information on how quickly the model processes and delivers predictions to evaluate its efficiency in urgent scenarios.

4.3. Results of the experiment

This section deals with the analysis of the results collected from the information provided in Fig. 7, which shows four graphs corresponding to each pollutant, comparing the real data with the predictions obtained from each model. Additionally, Fig. 8 presents the residual plots for each pollutant. The results for each pollutant are detailed in the following subsections, where the performance obtained in terms of precision and efficiency is presented with numerical values and discussed. Each pollutant is analyzed individually, identifying the best performing models from a selection of existing and newly developed models. The best results of each metric are highlighted in bold. Regarding execution times, no optimal result is highlighted as the simplest model is always the fastest, which corresponds to Decision Tree (DT).

Nitrogen dioxide

A number of conclusions emerge from Table 8 regarding the prediction results of the NO₂ pollutant using different machine-learning and deep-learning models. Firstly, it is observed that models based on convolutional neural networks (CNN) and recurrent networks (GRU, LSTM, BiLSTM) outperform the traditional models (DT, RF, MLP, GB, SVR) in terms of RMSE, R^2 , MAE, and Accuracy, indicating their ability to better capture the temporal and spatial characteristics of the data. Secondly, the M-CNN-BiLSTM model shows the best overall performance, with an RMSE of 7.64, an R^2 of 0.82, an MAE of 5.73, and an Accuracy of 0.94, surpassing other hybrid models combining CNN and LSTM. This suggests that the M-CNN-BiLSTM architecture is the most suitable for modeling the nonlinear relationship between the input variables and the target variable. Thirdly, the SVR model stands out as the only one achieving an MB close to zero, implying minimal bias in its predictions.

However, this model also has the longest training ($TIME_{T_r}$) and testing ($TIME_{T_e}$) times, limiting its practical applicability. Lastly, it should be noted that the GRUx2 and LSTMx2 models, using two stacked recurrent layers, do not show significant improvement over the single-layer GRU and LSTM models, suggesting that increased complexity does not necessarily lead to higher accuracy.

The heatmap of the results for NO₂ concentration predictions is shown in Fig. 9. It is important to note that the red color tones do not indicate hazardous concentration levels. The color scheme was chosen solely to highlight differences in predicted values.

Table 7

Quality criteria to evaluate the CALIOPE system based on statistical parameters (correlation coefficient, R ; mean bias, MB ; root mean square error, $RMSE$) taking into account previous studies and results (Arévalo, 2023).

Pollutant		MB ($\mu\text{g}/\text{m}^3$)	R^2 -	RMSE ($\mu\text{g}/\text{m}^3$)
O_3	Very Good	$ MB < 5$	$R^2 > 0.70$	$RMSE < 10$
	Good	$5 \leq MB < 10$	$0.50 < R^2 \leq 0.70$	$10 \leq RMSE < 20$
	Acceptable	$10 \leq MB < 20$	$0.30 < R^2 \leq 0.50$	$20 \leq RMSE < 30$
	Bad	$20 \leq MB < 30$	$0.10 < R^2 \leq 0.30$	$30 \leq RMSE < 40$
	Very Bad	$ MB \geq 30$	$R^2 \leq 0.10$	$RMSE \geq 40$
NO_2	Very Good	$ MB < 5$	$R^2 > 0.60$	$RMSE < 5$
	Good	$5 \leq MB < 10$	$0.40 < R^2 \leq 0.60$	$5 \leq RMSE < 15$
	Acceptable	$10 \leq MB < 20$	$0.20 < R^2 \leq 0.40$	$15 \leq RMSE < 25$
	Bad	$20 \leq MB < 30$	$0.10 < R^2 \leq 0.20$	$25 \leq RMSE < 35$
	Very Bad	$ MB \geq 30$	$R^2 \leq 0.10$	$RMSE \geq 35$
PM_{10}	Very Good	$ MB < 3$	$R^2 > 0.40$	$RMSE < 5$
	Good	$3 \leq MB < 5$	$0.30 < R^2 \leq 0.40$	$5 \leq RMSE < 15$
	Acceptable	$5 \leq MB < 10$	$0.20 < R^2 \leq 0.30$	$15 \leq RMSE < 25$
	Bad	$10 \leq MB < 20$	$0.10 < R^2 \leq 0.20$	$25 \leq RMSE < 35$
	Very Bad	$ MB \geq 30$	$R^2 \leq 0.10$	$RMSE \geq 30$
$PM_{2.5}$	Very Good	$ MB < 5$	$R^2 > 0.55$	$RMSE < 5$
	Good	$5 \leq MB < 10$	$0.40 < R^2 \leq 0.55$	$5 \leq RMSE < 15$
	Acceptable	$10 \leq MB < 20$	$0.25 < R^2 \leq 0.40$	$15 \leq RMSE < 25$
	Bad	$20 \leq MB < 30$	$0.10 < R^2 \leq 0.25$	$25 \leq RMSE < 35$
	Very Bad	$ MB \geq 30$	$R^2 \leq 0.10$	$RMSE \geq 35$

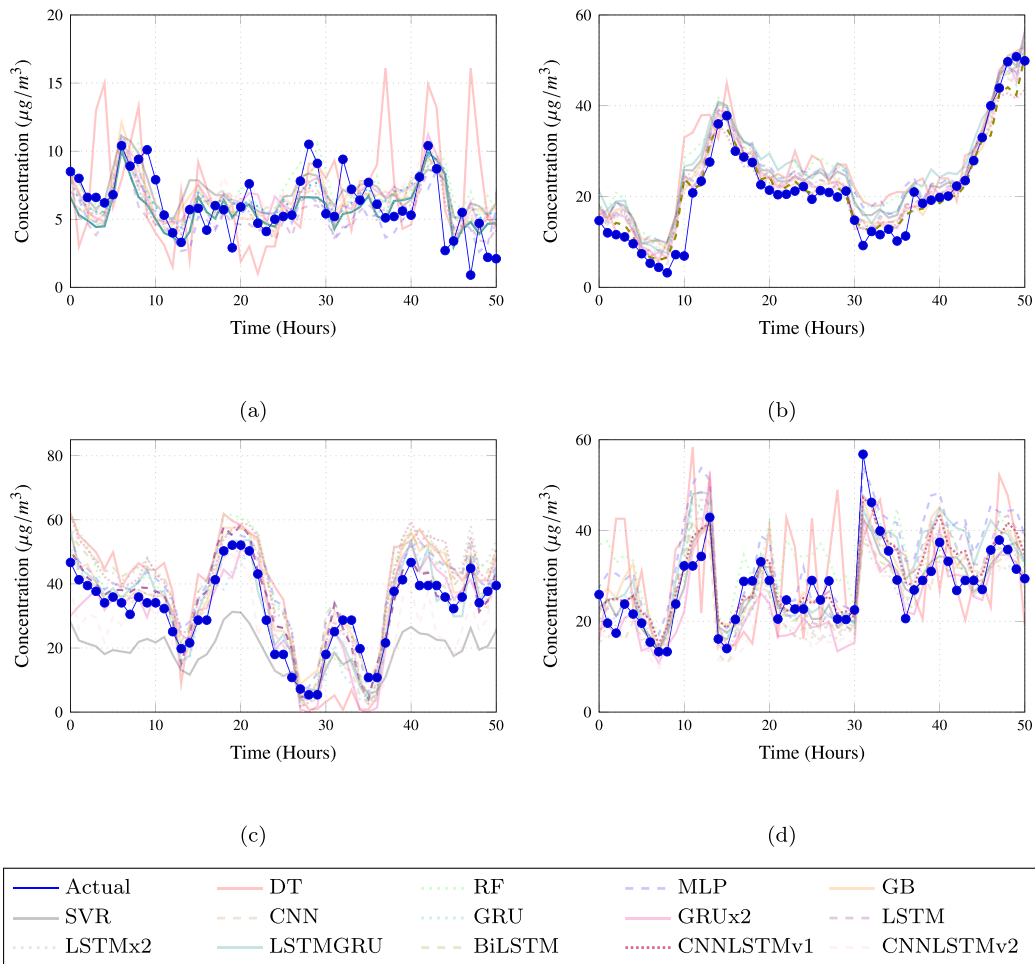


Fig. 7. The final prediction results are illustrated alongside the actual values in the four subfigures, all sharing the same legend. The subfigures represent the forecasting outcomes for (a) $PM_{2.5}$, (b) PM_{10} , (c) O_3 , and (d) NO_2 , respectively.

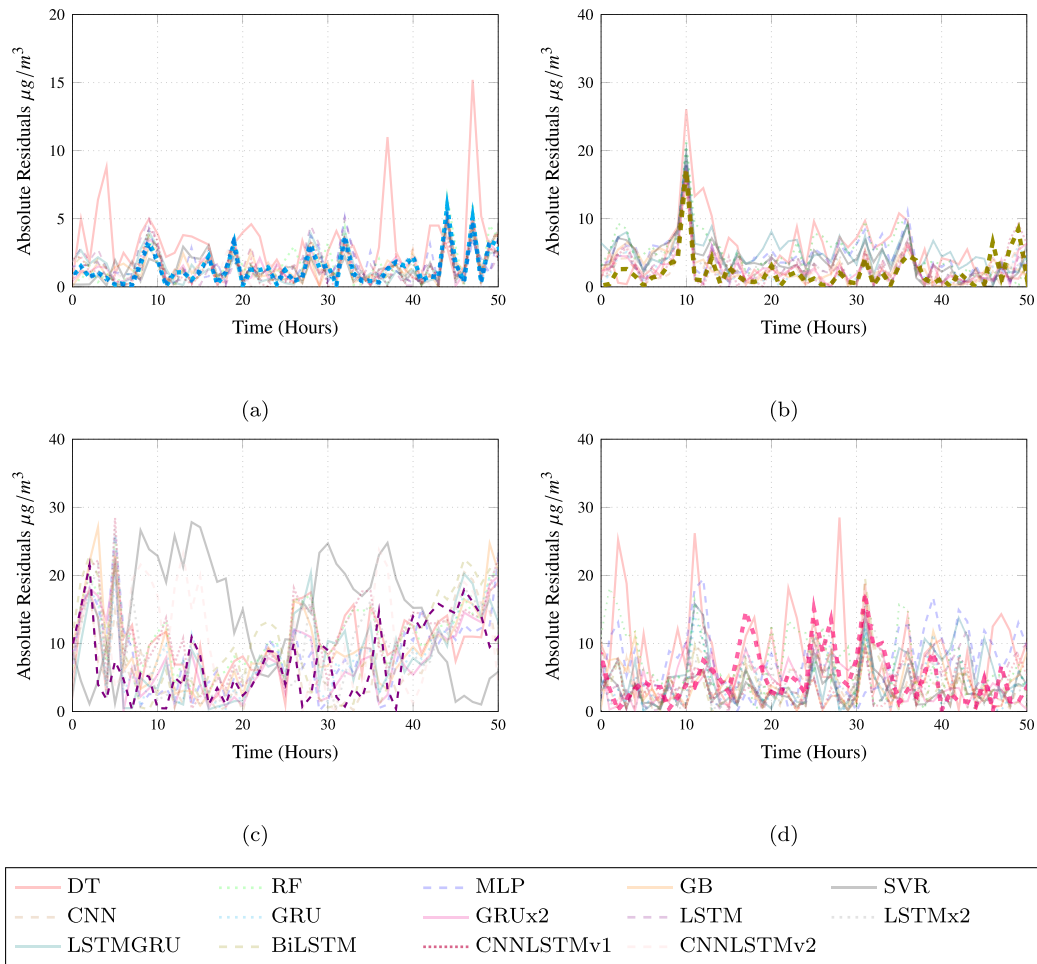


Fig. 8. Residual plots for (a) $PM_{2.5}$, (b) PM_{10} , (c) O_3 , and (d) NO_2 , respectively.

Table 8
Forecasting results for the pollutant NO_2 .

	RMSE	MB	R^2	MAE	Accuracy	$TIME_{Tr}$	$TIME_{Te}$
DT	12.97	1.62	0.49	9.27	0.9	29.88	–
RF	9.08	3.66	0.75	6.96	0.94	1554	0.51
MLP	12.47	8.9	0.52	10.43	0.88	9574	–
GB	8.43	2.89	0.78	6.53	0.94	447	–
SVR	7.92	0.9	0.8	5.93	0.94	530 807	1183
CNN	9.96	2.49	0.69	7.15	0.92	4183	3.29
GRU	9.8	3.58	0.71	7.56	0.91	8187	5.57
GRUx2	8.74	2.19	0.77	6.4	0.93	7537	4.34
LSTM	9.92	3.87	0.7	7.76	0.92	7144	5.04
LSTMx2	9.26	2.24	0.74	7.15	0.94	8498	6.24
LSTMGRU	10.78	5.00	0.64	8.55	0.91	4653	4.59
BiLSTM	9.47	0.46	0.73	7.17	0.93	4934	4.2
M-CNN-LSTM	10.18	0.87	0.68	7.85	0.94	7214	6.78
M-CNN-BiLSTM	7.64	1.68	0.82	5.73	0.94	11 531	7.87

Ozone

Table 9 provides the forecast results for the O_3 pollutant using various machine learning models, evaluated using different metrics. Here are some conclusions based on the results.

The LSTM model stands out as the best in terms of overall performance metrics, with the lowest RMSE (7.7), lowest MB (1.06), highest R^2 (0.86), lowest MAE (5.93), and high accuracy (0.92). This model demonstrates its ability to capture complex temporal relationships in ozone data.

The GB model also performs well, with an R^2 of 0.85, which is comparable to LSTM, and with much lower training and testing times.

This suggests that GB could be an attractive option if computational efficiency is prioritized.

Among recurrent neural network (RNN) models, LSTM and BiLSTM are the best in terms of performance. However, it is important to note that the CNN model also shows good performance, outperforming other recurrent neural networks like GRU, GRUx2, and LSTMGRU in several metrics.

In terms of computational efficiency, the GB model stands out for having significantly shorter training and testing times than models like LSTM. The CNN model also has a short training time.

In summary, the LSTM model remains the best choice in terms of accuracy. However, the GB model could be an attractive alternative if

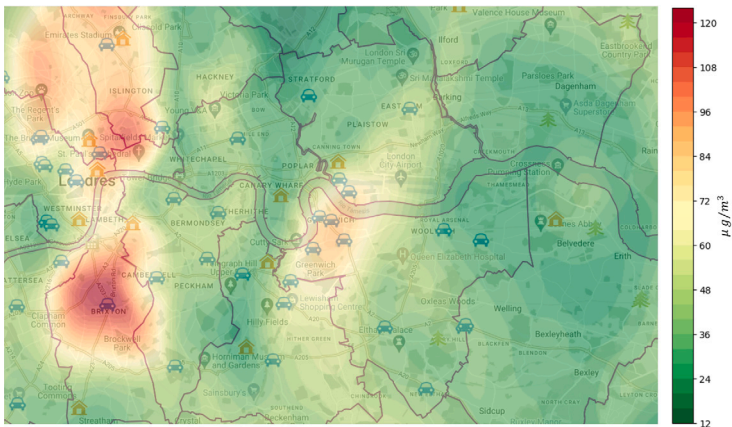


Fig. 9. Heatmap of NO₂ predictions.

Table 9
Forecasting results for the pollutant O₃.

	RMSE	MB	R ²	MAE	Accuracy	TIME _{Tr}	TIME _{Te}
DT	12.6	6.14	0.62	9.97	0.82	4.23	–
RF	10.06	5.41	0.76	8.05	0.84	409	0.24
MLP	8.58	3.18	0.82	6.84	0.9	970	–
GB	7.9	2.18	0.85	6.23	0.9	64	–
SVR	16.09	13.03	0.38	13.49	0.87	1121	82.7
CNN	8.5	3.22	0.83	6.29	0.93	355	2.08
GRU	8.72	2.31	0.82	6.48	0.92	1632	3.3
GRUx2	9.73	3.46	0.77	6.98	0.9	660	2.95
LSTM	7.7	1.06	0.86	5.93	0.92	1657	2.9
LSTMx2	9.42	1.68	0.79	7.33	0.88	1625	3.52
LSTMGRU	9.69	2.7	0.78	7.2	0.89	1053	3.0
BiLSTM	8.27	2.97	0.84	6.59	0.9	893	2.66
M-CNN-LSTM	9.86	2.28	0.77	7.18	0.9	1492	3.57
M-CNN-BiLSTM	9.73	1.74	0.77	7.1	0.88	1530	4.11

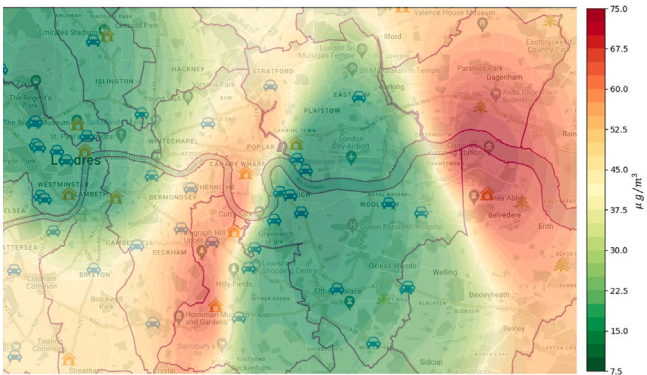


Fig. 10. Heatmap of O₃ predictions.

higher computational efficiency is sought, while the CNN model shows notable performance and may be preferred in situations where temporal complexity is not as critical as in LSTM.

The heatmap showing a set of O₃ concentration predictions for areas without deployed IoT devices is presented in Fig. 10. As with the previous pollutant, variations in color shades do not reflect hazard levels for O₃.

PM_{2.5}

According to these results in Table 10, it can be observed that the models based on recurrent neural networks (GRU, LSTMx2, LSTMGRU) achieve the best results in terms of RMSE, R², MAE, and Accuracy, indicating their ability to capture the temporal dynamics of the pollutant

and fit the observed data better. Among these models, the simple GRU has the lowest RMSE and the highest R², followed by the LSTMGRU, which combines a GRU layer with an LSTM layer. The M-CNN-BiLSTM and M-CNN-LSTM, which combine convolutional and recurrent layers, do not show significant improvement over the simple LSTM.

On the other hand, the MLP, CNN, DT, RF, GB, and SVR models perform worse than the recurrent models, although their performance is still acceptable. SVR has the lowest RMSE and the highest MAE among these models, suggesting it is more sensitive to extreme pollutant values. RF and GB yield similar results, with a shorter training time for GB. DT is the simplest and quickest model to train but also the least accurate in fitting the data.

Regarding the training and testing times, the recurrent models are computationally more expensive, especially when using two recurrent layers or when they are combined with convolutional layers. SVR also has a long testing time, limiting its applicability in real-time scenarios. The MLP, CNN, DT, RF, GB, and SVR models are faster to train and test but this comes at the expense of less accuracy.

In conclusion, the recurrent models are most suitable for predicting the PM_{2.5} pollutant due to their ability to model temporal dependencies in the series. However, these models require greater computational complexity and more parameters to optimize. Therefore, the trade-off between accuracy and efficiency should be evaluated when choosing the most appropriate model for each case study.

The heatmap of PM_{2.5} cross-location predictions is presented in Fig. 11, where color variations represent different particulate concentration levels.

PM₁₀

From Table 11, several conclusions can be drawn about the out-

Table 10
Forecasting results for the pollutant PM_{2.5}.

	RMSE	MB	R ²	MAE	Accuracy	TIME _{Tr}	TIME _{Te}
DT	4.33	0.39	0.71	3.01	0.89	4.16	–
RF	2.7	0.65	0.89	1.96	0.94	283	0.2
MLP	2.76	0.06	0.88	1.97	0.94	736	–
GB	2.69	0.8	0.89	1.93	0.93	70.7	–
SVR	2.65	0.47	0.89	1.88	0.94	2018	81.5
CNN	2.65	0.82	0.89	1.9	0.94	561	1.77
GRU	2.46	0.24	0.91	1.76	0.95	1160	2.7
GRUx2	2.74	0.58	0.89	1.9	0.94	1293	2.66
LSTM	2.83	0.19	0.88	1.99	0.94	1778	2.35
LSTMx2	2.58	0.38	0.9	1.79	0.94	2269	3.07
LSTMGRU	2.5	0.1	0.91	1.77	0.95	887	2.52
BiLSTM	2.71	0.29	0.89	1.85	0.94	1274	2.41
M-CNN-LSTM	2.67	0.98	0.89	1.97	0.94	2879	3.35
M-CNN-BiLSTM	2.68	0.48	0.89	1.87	0.94	4357	3.52

Table 11
Forecasting results for the pollutant PM₁₀.

	RMSE	MB	R ²	MAE	Accuracy	TIME _{Tr}	TIME _{Te}
DT	8.17	4.21	0.5	5.44	0.88	9.8	–
RF	6.36	4.02	0.7	4.77	0.9	962	0.18
MLP	5.52	3.53	0.77	4.33	0.92	562	–
GB	5.27	3.24	0.79	4.19	0.94	179	–
SVR	5.57	3.85	0.77	4.45	0.92	5765	72
CNN	4.66	2.1	0.84	3.33	0.93	819	0.99
GRU	4.59	2.28	0.84	3.36	0.94	2452	1.88
GRUx2	4.63	2.09	0.84	3.3	0.94	1538	1.68
LSTM	4.86	2.77	0.82	3.69	0.93	2536	1.76
LSTMx2	4.88	2.88	0.82	3.7	0.93	3750	1.88
LSTMGRU	5.82	3.92	0.74	4.63	0.9	1818	1.61
BiLSTM	4.26	0.99	0.86	2.89	0.94	1920	1.55
M-CNN-LSTM	4.9	2.84	0.82	3.78	0.94	1727	2.24
M-CNN-BiLSTM	4.63	1.43	0.84	3.19	0.94	4249	2.39

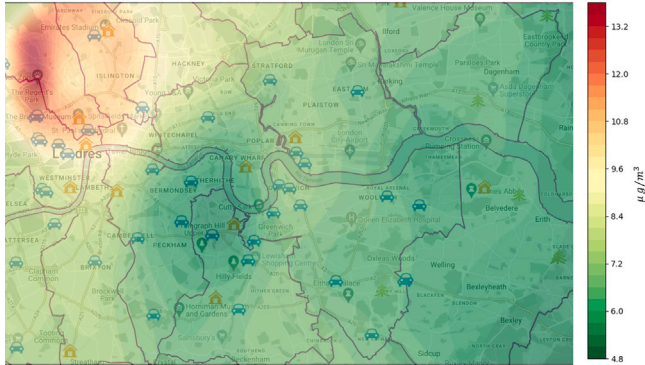


Fig. 11. Heatmap of PM_{2.5} predictions.

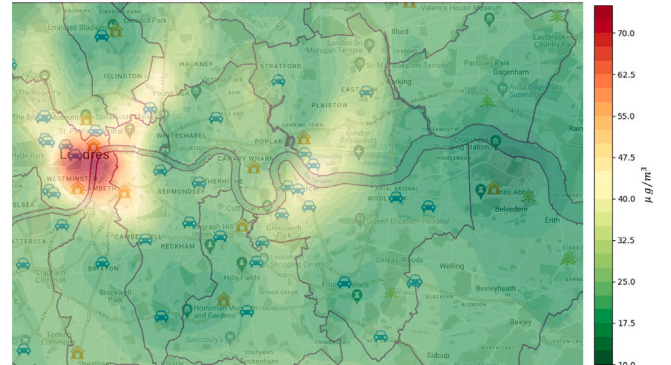


Fig. 12. Heatmap of PM₁₀ predictions.

comes of predicting the PM₁₀ pollutant using different machine-learning and deep-learning models.

Firstly, the models based on deep-learning techniques outperform traditional models such as Decision Tree (DT), Random Forest (RF), Multilayer Perceptron (MLP), Gradient Boosting (GB), and Support Vector Machine (SVR). This is because RNNs are capable of capturing the temporal dependencies of the data series and learning complex patterns. Among the RNN architectures, the bidirectional LSTM network (BiLSTM) stands out, achieving the lowest RMSE, the lowest MAE, the highest R², and the highest accuracy. The BiLSTM network leverages information from the past and the future to make a more precise prediction.

Secondly, combining convolutional neural networks (CNNs) and RNNs improves the performance of pure RNNs, especially in the case of the M-CNN-BiLSTM network, which has a more complex structure than M-CNN-LSTM. The M-CNN-BiLSTM network uses a CNN layer to

extract relevant features from the time series and then pass them to an LSTM layer to model temporal dependencies. Thirdly, the training and testing times vary considerably among the models, with traditional models being the fastest and SVR-based models being the slowest. RNN models have an intermediate time, depending on the number of layers and hidden units. The slowest model is the M-CNN-BiLSTM due to its increased complexity. Finally, we conclude that deep-learning models are an effective and promising tool for air quality prediction, especially the BiLSTM model, which offers a good balance between accuracy and speed.

The heatmap of PM₁₀ predictions is shown in Fig. 12. Note that areas shaded in red or yellow do not necessarily indicate hazardous pollutant levels.

A summary of the results obtained for each pollutant is provided in Table 12. It can be observed that different pollutants were best predicted by different models. While training these models can be

Table 12
Summary of the best performing models for each pollutant.

Pollutants	Best model
PM _{2.5}	GRU
PM ₁₀	BiLSTM
NO ₂	M-CNN-BiLSTM
O ₃	LSTM

resource-intensive, the inference phase has a relatively low computational cost, facilitating scalability and making them suitable for integration into city-scale applications that require predictions across many geographic points simultaneously. Despite this scalability, the proposed approach is not intended for deployment on low-resource edge devices, nor would such deployment be appropriate for our use case. Since our objective is to perform cross-location predictions, edge inference is not feasible. This is because individual devices lack access to the broader context provided by data from other sensors and are not capable of processing and managing such large-scale, distributed information. An alternative could be the use of federated learning techniques, where each device receives a pre-trained model and contributes to training in a distributed manner. However, our approach targets air quality prediction in areas without any deployed IoT devices. Therefore, even in a federated scenario, these edge devices would not be present in the target locations, and such an approach would still fall outside the scope of our intended use case.

Regarding scalability, the proposed architectures (CNN-LSTM and CNN-BiLSTM) are computationally efficient during inference, making them suitable for integration into real-time systems or city-scale applications that require predictions across many geographic points simultaneously.

While training deep learning models can be resource-intensive, once trained, the inference phase — required for deployment — has relatively low computational cost, especially when batch predictions are performed. This makes the approach viable for deployment on centralized servers or edge computing devices with moderate resources.

5. Conclusion

In conclusion, the article addresses the challenge of predicting air quality in urban populations based on geographical location. Artificial intelligence models are used to efficiently achieve this objective. The advantages, disadvantages, virtues, and variability of the behavior of each pollutant are explored, focusing on the context of publicly available data from the city of London.

The comparative study includes: (i) classic machine learning models and neural networks designed with convolutional and/or recurrent layers and (ii) new models combining CNN and LSTM techniques. Metrics such as MB, RMSE, MAE, and R² have been used to evaluate accuracy, along with qualitative evaluation using the Accuracy metric and considerations about computational efficiency through training and testing times.

The importance of these execution times is highlighted, especially when real-time application in specific areas is sought, which requires predicting multiple areas to adequately visualize and estimate the environmental situation. The experimental tests presented in this paper involved a large number of simultaneous predictions (which could be considered a worst-case scenario), even though real-time applications would require forecasts for a smaller, user-specified set of geographical points. Nevertheless, we will explore techniques such as model pruning, quantization, and knowledge distillation in future work to further improve the computational efficiency of our model.

In the assessment, a careful selection of subsets of relevant factors for each pollutant is performed, allowing the best sets of predictor variables for each one to be identified. It is thus confirmed that pollutants

do not behave in the same way, which is reflected in the diversity of effective models for each case.

Specifically, we have found that the combination of convolutional and recurrent layers is more efficient for nitrogen dioxide, while a single LSTM recurrent layer is optimal for ozone. For particles smaller than 2.5 micrograms per cubic meter, models with a single GRU recurrent layer show greater efficiency. Finally, in the case of particles smaller than 10 micrograms per cubic meter, the BiLSTM model shows the best performance, highlighting the importance of considering both retrospective and prospective evolution when making predictions. These insights can assist urban planners and policymakers in determining the optimal locations for parks, hospitals, schools, and other infrastructure to improve air quality and ensure that vulnerable populations benefit from the healthiest environments. Additionally, they can be used to regulate traffic in real-time, mitigating air pollution spikes as efficiently as possible. In the event of pollution surges, this approach can help identify pollution sources, particularly in industrial areas. These exceptional events would not significantly compromise model performance, as the cross-location approach proposed in this work allows anomalies to be detected by other IoT monitoring stations. Consequently, reliable values can still be inferred for areas without deployed devices.

It is also important to note that, although our study focuses on the city of London, we acknowledge that location-specific dynamics may limit the direct applicability of the trained models to other urban environments without further adaptation. Air pollution patterns can vary depending on a city's urban layout. Industrial zones, high-traffic roads and intersections, specific meteorological phenomena, and even topographical features can give rise to localized areas with elevated pollution levels. These factors influence both the generation and dispersion of pollutants, which may, in turn, affect the relationships learned by the model. Nevertheless, the proposed methodology is designed to be generalizable in terms of both its architecture and training strategy. While model parameters may need to be retrained or fine-tuned using data from a different location, the overall framework — including the integration of spatial and temporal patterns through CNN and LSTM components — remains broadly applicable.

Given these considerations, future work will focus on evaluating the proposed models in other cities. However, it is also crucial to acknowledge the potential biases introduced by AI in IoT data deployment, as well as the associated ethical and policy implications. IoT sensors are often unevenly distributed, with a higher concentration in city centers and fewer devices in peripheral areas. This imbalance can lead to skewed predictions in under-monitored regions. Nevertheless, addressing predictions in areas with limited sensor coverage is a key focus of this study. Furthermore, it is important to note that the approach proposed in this paper may face limitations in exceptional cases of pollution outbursts occurring in areas without deployed air quality monitoring devices.

Furthermore, the use of AI in air quality monitoring must ensure responsible data collection practices that respect citizens' privacy. The results generated by these models should not be used to directly impose fines on individuals or industries but rather to provide an initial assessment that requires further expert evaluation. Ultimately, AI serves as a decision-support tool, but final determinations should be made by human stakeholders.

Lastly, also as part of our future work, we will develop a model for short-term predictions at individual monitoring stations. This approach will focus on forecasting air quality values for the next few hours at specific sensor locations. Unlike the model presented in this paper, this task will require a more compact and continuous dataset, minimizing temporal gaps, as recurrent neural networks (RNNs) could play a crucial role in effectively handling time-series data. Achieving highly accurate short-term predictions could significantly enhance the original problem by serving as a powerful data augmentation technique. This would not only increase the number of available samples but also improve dataset consistency by ensuring that data points are temporally closer together.

Once these objectives are met, integrating both zonal and temporal predictions could further refine air quality forecasting. Even without prior improvements to the original problem, this combination could yield valuable insights and represent a significant advancement in the field. Furthermore, we will also explore alternative learning paradigms to enhance performance, such as Variance-Constrained Local-Global Modeling (Zhang et al., 2023) and Integrated Multiple Kernel Learning (IMKL) (Zhang et al., 2020).

CRedit authorship contribution statement

Fernando Illescas-Martinez: Software, Investigation, Data curation, Writing – original draft. **Laura Garcia:** Methodology, Conceptualization, Data curation, Writing – original draft, Writing – review & editing, Visualization. **Antonio-Javier Garcia-Sanchez:** Methodology, Conceptualization, Data curation, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration, Funding acquisition. **Rafael Asorey-Cacheda:** Conceptualization, Resources, Supervision, Funding acquisition. **Joan Garcia-Haro:** Validation, Resources, Supervision, Writing – original draft, Writing – review & editing, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the grants PID2020-116329GB-C22, TED2021-129336B-I00, and FJC2021-047073-I funded by MCIN/AEI/10.13039/501100011033 and by the European Union NextGenerationEU/PRTR. This work was also supported by the grant PCI2024-153485 funded by MICIU/AEI/10.13039/501100011033 and by the European Union. This research was also contextualized to DAIMon, a cascade funding action deriving from the Horizon Europe project aerOS, funded by the European Commission under grant number 101069732.

Data availability

Data will be made available on request.

References

- Agbehadj, I. E., & Obagbuwa, I. C. (2024). Systematic review of machine learning and deep learning techniques for spatiotemporal air quality prediction. *Atmosphere*, 15(11), 1352.
- AlShehhi, A., & Welsch, R. (2023). Artificial intelligence for improving nitrogen dioxide forecasting of abu dhabi environment agency ground-based stations. *Journal of Big Data*, 10(1), 92.
- Arévalo, G. (2023). Evaluación del sistema de pronóstico de calidad del aire CALIOPE en españa 2013. https://www.bsc.es/projects/earthscience/visor/bases_datos/image_viewer/docs/20140513_Informe_Evaluacion_Pronostico_CALIOPE_2013.pdf. (Online; Accessed 15 May 2023).
- Aslam, N., Rustam, F., Lee, E., Washington, P. B., & Ashraf, I. (2022). Sentiment analysis and emotion detection on cryptocurrency related tweets using ensemble LSTM-GRU model. *Ieee Access*, 10, 39313–39324.
- Awad, M., Khanna, R., Awad, M., & Khanna, R. (2015). Support vector regression. *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*, 67–80.
- Ayuntamiento de Madrid (2023). Portal de calidad del aire. <https://airedemadrid.madrid.es/portales/calidadaire/es/Actuaciones-municipales/Sistema-integral-de-calidad-del-aire/Informacion/Indice-de-Calidad-del-Aire/?vgnnextfmt=default&vgnnextoid=303d635a41187710VgnVCM1000001d4a900aRCRD&vgnnextchannel=ec7e471c5c503710VgnVCM1000008a4a900aRCRD>. (Online; Accessed 15 May 2023).
- Babu Saheer, L., Bhasya, A., Maktabdar, M., & Zarrin, J. (2022). Data-driven framework for understanding and predicting air quality in urban areas. *Frontiers in Big Data*, 5, Article 822573.
- Bekkar, A., Hssina, B., Douzi, S., & Douzi, K. (2021). Air-pollution prediction in smart city, deep learning approach. *Journal of Big Data*, 8(1), 1–21.
- Calo, S., Bistaffa, F., Jonsson, A., Gómez, V., & Viana, M. (2024). Spatial air quality prediction in urban areas via message passing. *Engineering Applications of Artificial Intelligence*, 133, Article 108191.
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.
- Dairi, A., Harrou, F., Khadraoui, S., & Sun, Y. (2021). Integrated multiple directed attention-based deep learning for improved air pollution forecasting. *IEEE Transactions on Instrumentation and Measurement*, 70, 1–15.
- Dong, J., Zhang, Y., & Hu, J. (2024). Short-term air quality prediction based on EMD-transformer-bilstm. *Scientific Reports*, 14(1), 20513.
- Environmental Research Group, Imperial College London (2025). London air quality network — londonair.org.uk. <https://www.londonair.org.uk/LondonAir/Default.aspx>. (Accessed 07 February 2025).
- Estalayo, I., Del Ser, J., Osaba, E., Bilbao, M. N., Muhammad, K., Gálvez, A., & Iglesias, A. (2019). Return, diversification and risk in cryptocurrency portfolios using deep recurrent neural networks and multi-objective evolutionary algorithms. In *2019 IEEE congress on evolutionary computation* (pp. 755–761). IEEE.
- García, L., García-Sánchez, A.-J., Asorey-Cacheda, R., García-Haro, J., & Zúñiga Cañón, C.-L. (2022). Smart air quality monitoring IoT-based infrastructure for industrial environments. *Sensors*, 22(23), 9221.
- Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., et al. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, 77, 354–377.
- Hameed, S., Islam, A., Ahmad, K., Belhaouari, S. B., Qadir, J., & Al-Fuqaha, A. (2023). Deep learning based multimodal urban air quality prediction and traffic analytics. *Scientific Reports*, 13(1), 22181.
- Heydari, A., Majidi Nezhad, M., Astiaso Garcia, D., Keynia, F., & De Santoli, L. (2022). Air pollution forecasting application based on deep learning model and optimization algorithm. *Clean Technologies and Environmental Policy*, 1–15.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.
- James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2013). *An introduction to statistical learning*: vol. 112, Switzerland: Springer Cham.
- Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, 31(3), 685–695.
- Jayaraman, S., & Abirami, S. (2025). Enhancing urban air quality prediction using time-based-spatial forecasting framework. *Scientific Reports*, 15(1), 4139.
- Kingsford, C., & Salzberg, S. L. (2008). What are decision trees? *Nature Biotechnology*, 26(9), 1011–1013.
- Le, V.-D., Bui, T.-C., & Cha, S.-K. (2020). Spatiotemporal deep learning model for citywide air pollution interpolation and prediction. In *2020 IEEE international conference on big data and smart computing (bigComp)* (pp. 55–62). IEEE.
- Lu, Y., & Li, K. (2023). Multistation collaborative prediction of air pollutants based on the CNN-bilstm model. *Environmental Science and Pollution Research*, 30(40), 92417–92435.
- Mampitiya, L., Rathnayake, N., Leon, L. P., Mandala, V., Azamathulla, H. M., Shelton, S., Hoshino, Y., & Rathnayake, U. (2023). Machine learning techniques to predict the air quality using meteorological data in two urban areas in Sri Lanka. *Environments*, 10(8), 141.
- Marinov, M. B., Topalov, I., Gieva, E., & Nikolov, G. (2016). Air quality monitoring in urban environments. In *2016 39th International spring seminar on electronics technology* (pp. 443–448). IEEE.
- Mitreska Jovanovska, E., Batz, V., Lameski, P., Zdravetski, E., Herzog, M. A., & Trajkovic, V. (2023). Methods for urban air pollution measurement and forecasting: Challenges, opportunities, and solutions. *Atmosphere*, 14(9), 1441.
- Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in Neuroinformatics*, 7, 21.
- Popescu, M.-C., Balas, V. E., Perescu-Popescu, L., & Mastorakis, N. (2009). Multilayer perceptron and neural networks. *WSEAS Transactions on Circuits and Systems*, 8(7), 579–588.
- Pražnikar, Z., & Pražnikar, J. (2012). The effects of particulate matter air pollution on respiratory health and on the cardiovascular system. *Slovenian Journal of Public Health*, 51(3), 190–199.
- 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.
- Rigatti, S. J. (2017). Random forest. *Journal of Insurance Medicine*, 47(1), 31–39.
- Schürholz, D., Kubler, S., & Zaslavsky, A. (2020). Artificial intelligence-enabled context-aware air quality prediction for smart cities. *Journal of Cleaner Production*, 271, Article 121941.
- Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), 2673–2681.
- Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2019). The performance of LSTM and BiLSTM in forecasting time series. In *2019 IEEE international conference on big data (big data)* (pp. 3285–3292). IEEE.

- Wang, X., Zhang, S., Chen, Y., He, L., Ren, Y., Zhang, Z., Li, J., & Zhang, S. (2024). Air quality forecasting using a spatiotemporal hybrid deep learning model based on VMD-GAT-BiLSTM. *Scientific Reports*, 14(1), 17841.
- Wu, C.-l., He, H.-d., Song, R.-f., Zhu, X.-h., Peng, Z.-r., Fu, Q.-y., & Pan, J. (2023). A hybrid deep learning model for regional O₃ and NO₂ concentrations prediction based on spatiotemporal dependencies in air quality monitoring network. *Environmental Pollution*, 320, Article 121075.
- Xie, J., Liu, F., Liu, S., & Jiang, X. (2025). An approach to spatiotemporal air quality prediction integrating SwinLSTM and kriging methods. *Sustainability*, 17(7), 2918.
- Zaremba, W., Sutskever, I., & Vinyals, O. (2014). Recurrent neural network regularization. arXiv preprint [arXiv:1409.2329](https://arxiv.org/abs/1409.2329).
- Zhang, J., Li, Y., Li, Q., & Xiao, W. (2023). Variance-constrained local-global modeling for device-free localization under uncertainties. *IEEE Transactions on Industrial Informatics*, 20(4), 5229–5240.
- Zhang, J., Li, Y., & Xiao, W. (2020). Integrated multiple kernel learning for device-free localization in cluttered environments using spatiotemporal information. *IEEE Internet of Things Journal*, 8(6), 4749–4761.
- Zhou, J., Cao, Y., Wang, X., Li, P., & Xu, W. (2016). Deep recurrent models with fast-forward connections for neural machine translation. *Transactions of the Association for Computational Linguistics*, 4, 371–383.