GT-LSTM: Integrating High-Resolution Particulate Matter Data for Urban Air Quality Forecasting

Maryam Rahmani, Suzanne Crumeyrolle, Nadège Martiny, Romain Rouvoy

GT-LSTM: Integrating High-Resolution Particulate Matter Data for Urban Air Quality Forecasting

Maryam Rahmani[1], Suzanne Crumeyrolle[2], Nadège Martiny[3], and Romain Rouvoy[1]

[1] Univ. Lille, Inria, CNRS, UMR 9189 CRIStAL, France
[2] Univ. Lille, CNRS, UMR 8518 LOA, France
[3] Univ. Bourgogne Europe, CNRS, UMR 6282 Biogéosciences, France

Abstract. Air pollution remains a critical environmental and public health challenge in urban areas, which requires accurate and efficient predictive models to mitigate its impact. This study introduces a novel spatiotemporal model, *Graph Temporal LSTM* (GT-LSTM), which integrates *Machine Learning* (ML) techniques to forecast air pollution levels, with a focus on *Particulate Matter* $(PM)_{2.5}$ concentrations. By combining *Graph Convolutional Network* (GCN) to capture spatial dependencies and *Long Short-Term Memory* (LSTM) to model temporal patterns, the proposed framework provides precise and localized predictions in urban and suburban regions.

Our analysis demonstrates the competitive predictive capabilities of the model, achieving high *Coefficient of determination* (R^2) and low error values, highlighting its robustness in correlating predicted and observed pollutant levels. The GT-LSTM model effectively incorporates historical data, neighboring influences, and local pollution sources, allowing reliable short- and long-term forecasts, even in data-short environments.

In addition to its predictive accuracy, the model prioritizes computational efficiency and scalability, using cost-effective sensor networks to expand coverage and reduce the dependence on traditional data sources. By offering fine-grained insights into air quality patterns, this approach supports real-time monitoring, long-term planning, and proactive decision-making, benefiting policymakers and urban residents alike. This study underscores the transformative potential of spatiotemporal modeling and ML techniques in enhancing air pollution monitoring systems, ultimately contributing to improved air quality management and public health outcomes.

Keywords: Graph Neural Networks · Temporal LSTM · Spatiotemporal Forecasting · Air Pollution Prediction · Air Quality Sensor Networks · Particulate Matter Modeling · Air Quality Monitoring · Deep Learning for Environmental Data · Dynamic Graph Learning · Time Series Forecasting

1 Introduction

As global climate change accelerates, its consequences on environmental and public health grow increasingly severe. Among these, air pollution emerges, not only as a contributor to climate change, but also as a critical standalone threat to human health and ecological stability [1,22].

Air pollution encompasses a variety of harmful substances, including *Particulate Matter*s (PMs), *Nitrogen Oxides* (NOx), *Sulfur Dioxide* (SO_2), *Ozone* (O_3), and *Volatile Organic Compounds* (VOCs). In particular, PM, consisting of fine particles like $PM_{2.5}$ and PM_{10}, has been strongly linked to respiratory and cardiovascular diseases [15]. Given its dual role in climate dynamics and health outcomes, improving air pollution forecasting is essential for sustainable urban planning, real-time policy interventions, and public well-being.

Identifying the main pollutant sources is crucial for accurate forecasting and for supporting strategic planning in various sectors, from large-scale industries to small-scale agricultural enterprises [11].

This proactive approach enables the implementation of effective measures to mitigate environmental impacts and promote sustainable development [3]. However, the need for accurate, high-resolution air pollution forecasting is particularly pressing in urban areas, where it is essential to address both temporal forecasting and the spatial granularity of forecasting across different site locations within the city, especially for pollutants such as $PM_{2.5}$. Conventional models often rely on data from a limited number of monitoring stations, which may fail to capture the complex dynamics of air pollution in densely populated areas [16].

Air quality forecasting often involves the use of various *Artificial Intelligence* (AI) methods, including *Long Short-Term Memory*s (LSTMs) networks [28], *support vector machines*, and *random forests*, which are applied to multidimensional time series data [25]. These methods, while differing in approach, have been shown to improve the accuracy and efficiency of predictions. However, the effectiveness of each method can vary depending on the dataset and problem context, with LSTMs often performing particularly well in capturing temporal dependencies.

While temporal analysis is important, integrating spatiotemporal considerations into forecasting models is equally critical for a comprehensive understanding of air quality [10]. Traditional approaches, including physical or chemical models, fail to account for the spatial variability of pollutant concentrations across different urban neighborhoods [19]. Air pollution impacts vary significantly between areas due to factors like traffic density, industrial, residential and agriculture activities, and land use patterns. Therefore, a more nuanced approach is needed to provide detailed information on pollutant sources and urban topology [30].

Incorporating spatiotemporal data into air quality forecasting models enables detailed analysis of pollution trends, identification of emission hotspots, and assessment of exposure risks for vulnerable populations. By leveraging data from multiple monitoring stations and employing advanced *Machine Learning*

(ML) techniques—particularly graph-based models such as glsplgnn and *Graph Attention Network*s (GATs)—these models can effectively capture complex interactions between local emissions, meteorological factors, and geographic features, thereby improving prediction accuracy [7].

This research presents a spatiotemporal forecasting framework that integrates ML-based graph modeling with air quality data, focusing on the city of Dijon, France. The approach delivers localized forecasts to support targeted interventions and data-driven policy decisions for air quality management [12].

To improve the accuracy of short-term $PM_{2.5}$ forecasting, we introduce a hybrid model that combines *Graph Convolutional Network*s (GCNs) and LSTMs within a temporal attention framework. By incorporating spatial relationships from sensor networks into our previously validated LSTM-based architecture, the model captures local pollution dynamics more effectively—especially in suburban areas with sparse monitoring infrastructure. It enables flexible prediction windows ranging from one hour to over 24 hours.

The rest of this paper is organized as follows: section 2 reviews related work, section 3 and section 4 describe the methodology and dataset, section 5 presents experimental results, and section 6 concludes with key findings and future research directions.

2 Related Works

Over the past two decades, significant research has focused on forecasting air quality using ML algorithms. The release of pollutants, such as PMs, NOx, SO2, O3, and VOCs, into the atmosphere raises serious risks to human health and ecosystems [23]. Accurate and timely air pollution forecasts are, therefore, crucial for mitigating these adverse effects, making this a major concern for researchers.

Traditional statistical models, such as regression and time series analysis, have been used to predict air pollution levels. However, these methods often struggle to capture the complex, nonlinear relationships inherent in air quality data [29]. To overcome these limitations, ML techniques, including *Support Vector Machines* (SVMs), *Artificial Neural Networks* (ANNs), and *Recurrent Neural Network*s (RNNs), have been explored [20]. Although these models demonstrate strong capabilities in capturing temporal patterns, they often fail to account for spatial dependencies [20,14].

In particular, Belavadi *et al.* [2] developed a scalable architecture for real-time air quality monitoring and forecasting by integrating wireless sensor networks with government-provided open data. While the system demonstrated strong potential, it also highlighted challenges posed by temporal variability in air quality across regions. To address these challenges, subsequent studies, such as [5], employed LSTM models to capture temporal dynamics and emphasized the role of meteorological factors in enhancing predictive accuracy.

Recognizing the limitations of temporal-only models, researchers have increasingly focused on integrating spatial dependencies through spatiotemporal

architectures. Some methods combine *Convolutional Neural Network*s (CNNs) with LSTMs to jointly model spatial and temporal patterns [26]. For example, Gilik *et al.* [9] proposed a CNN-LSTM model to forecast pollutant concentrations across urban locations and enable model transferability between cities. Similarly, Unjin Pak *et al.* [24] applied this approach to predict daily average $PM_{2.5}$ concentrations, though the model's complexity and data dependency may hinder broader applicability.

Zhang Qi *et al.* [31] introduced Deep-AIR, a hybrid CNN-LSTM model that incorporates domain-specific features like street canyon effects to improve city-scale pollution forecasting. While Deep-AIR yields promising results, its reliance on granular urban data may limit its generalizability to finer spatial scales.

Graph-based models have gained popularity for their ability to capture spatial dependencies. GCNs represent monitoring stations as graph nodes, with edges defined by geographic proximity or pollution similarity [4]. Hofman *et al.* [13] demonstrated that combining GCNs with RNNs improves forecasting accuracy by modeling both spatial and temporal structures. Their work also explored mobile sensor data to generate high-resolution air quality maps, outperforming traditional interpolation methods. Despite these advances, such models remain sensitive to data quality, sensor density, and regional variability.

To further address spatiotemporal complexities, Liang *et al.* [8] proposed the *Multi-scale Spatio-Temporal Graph Convolution Network* (MST-GCN), which effectively captures multi-level spatial correlations and long-term temporal dependencies. Although MST-GCN surpasses baseline models and addresses multi-source data challenges, its sophisticated architecture increases computational costs.

A more recent development is the *Spatiotemporal Graph Convolutional Recurrent Neural Network* (Spatiotemporal GCRNN) [17], which integrates GCNs with RNNs in a lightweight architecture. While the Spatiotemporal GCRNN is smaller and more efficient than prior models such as ConvLSTM [18], it still relies heavily on comprehensive and diverse data sources.

To overcome limitations in existing forecasting models—such as their reliance on dense urban datasets or computationally intensive methods—this study emphasizes *computational efficiency* and *scalability*. Leveraging low-cost in-situ sensors, the proposed framework models both spatial and temporal dependencies using a lightweight GCN-LSTM architecture. This design reduces dependence on costly traditional data sources and supports fine-grained, subarea-level forecasting in regions with sparse sensor coverage.

3 Method

The primary objective of this research is to forecast air pollution levels within a specific time frame by using historical environmental data from various areas within a city. Specifically, our focus is on predicting $PM_{2.5}$ pollutant levels for different areas by analyzing characteristics of spatio-temporal datasets in urban regions.

The problem statement is defined using a weighted graph $G = (N, E)$ to represent the topological structure of the city. Each measurement station is treated as a node, where N represents the set of station nodes $N = [N_1, N_2, ..., N_n]$, n is the total number of nodes. The edge set, denoted as E, defines connections between nodes, illustrating how one node is linked to another. We represent this set of edges with a special adjacency matrix. The adjacency matrix A, denoted as $A \in R^{n \times n}$, visually represents the connections between nodes based on the correlation coefficients obtained from the topology of the city.

Pearson's algorithm is employed to compute the correlation coefficient between node pairs, constructing the adjacency matrix A. As shown in Equation 1, X and Y represent the measurements from two sensor nodes, Cov signifies the covariance, and σ represents the standard deviation. The key advantage of using Pearson correlation lies in its ability to quantify the linear relationship between measurements, capturing how similarly nodes behave over time. Unlike distance-based measures, Pearson correlation can reveal meaningful connections between nodes that may not be geographically close but exhibit similar pollution patterns due to shared environmental factors. This approach enhances the model's ability to capture complex spatial dependencies beyond mere spatial proximity.

$$P_{X,Y} = \frac{Cov(X, Y)}{\sigma_X . \sigma_Y} \tag{1}$$

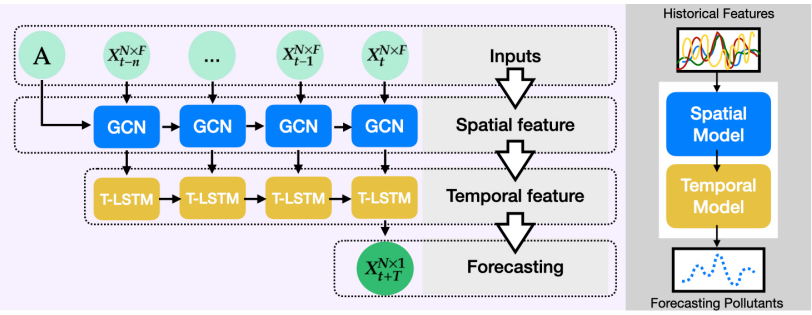

Fig. 1: Overview of *Graph Temporal LSTM* (GT-LSTM) comprising Spatial and Temporal Models. GCN and T-LSTM, respectively, are shown as the main components

Air pollution information within the network is represented by a feature matrix, denoted as $\mathbf{X}$. This matrix has dimensions $N \times F \times t$, where N is the number of nodes (monitoring stations), F is the number of features (e.g., pollutant concentrations, meteorological data), and t is the time step. By structuring the data in this way, we can effectively analyze and model the relationships between air pollution levels, node attributes, and temporal patterns within the network.

Spatiotemporal air pollution forecasting involves learning a mapping function f that predicts future air pollution levels given a stationary network topology G and a feature matrix $\mathbf{X}$. The goal is to predict air pollution levels for the next T time steps. The output, denoted as $\mathbf{X}_{t+T}^{N\times M}$, represents the predicted air pollution levels for all N nodes over the next T time steps. In this study, we focus on predicting $PM_{2.5}$ concentrations, setting $M = 1$. However, the proposed framework can be extended to multiple pollutants, treating M as the number of output variables, thereby enabling a multi-task learning approach.

The relationship depicted in our work is formalized by Equation 2:

$$X_{t+1}^{N\times 1}, ..., X_{t+T}^{N\times 1} = f(G, X_{t-n}^{N\times F}, ..., X_{t-1}^{N\times F}, X_{t}^{N\times F}) \tag{2}$$

Figure 1 illustrates the implementation details of our framework, incorporating the parameters described earlier. The figure presents the architecture and modules of our model, demonstrating how the various elements interact to forecast future air pollutant levels. Each component will be elaborated upon in the following sections.

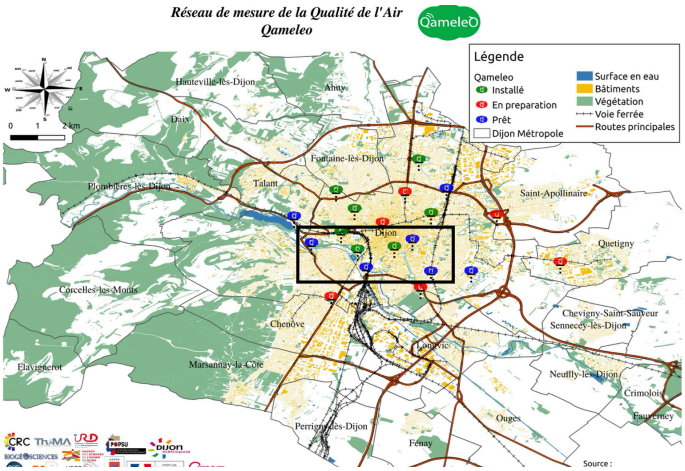

Fig. 2: Locations of Air Pollution Monitoring Stations in Dijon. The blue circles correspond to the active stations used in this study. [21]

3.1 Graph Temporal LSTM (GT-LSTM)

To effectively capture both spatial and temporal dependencies among monitoring stations, we propose a novel spatiotemporal model termed GT-LSTM. Our study focuses on four monitoring stations within the air pollution network of Dijon, France: Canal, Hoche, Carnot, and Janin, designated as node 1 to node 4, respectively. Figure 2 provides a visual representation of the spatial distribution

of the sensors deployed or soon to be deploy in this city. Data from the four blue sites in the black box are included in this study.

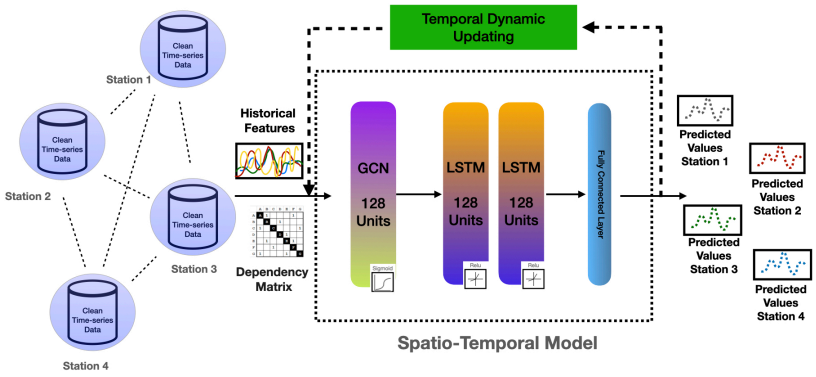

Fig. 3: Proposed spatiotemporal model. The model incorporates GCN blocks for capturing spatial features, LSTM blocks for capturing temporal features, temporal dynamic updating blocks, input data, a dependency matrix representing spatial relationships, and the predicted outputs.

3.2 Spatial Model

When it comes to predicting air pollution in an urban area, efficient topological resolution is essential. Relying on a single measurement in a city is often insufficient, particularly in larger cities. Therefore, to ensure accurate and comprehensive prediction, it is crucial to consider multiple measurements distributed throughout the urban area, motivating this study to have sub-zones for monitoring. To extract spatial data from a graph using a neural network, we employ GCN, which is an extension of CNN specifically designed to handle diverse graph-structured data [6]. In GCN, the process involves multiplying the input neurons by a set of weights, known as *filters* or *kernels*. These filters act as a sliding window across the entire data, allowing GCN to learn the characteristics of neighboring nodes within the graph. As described earlier, we effectively capture and learn spatial information from the graph structure. A GCN takes:

- an input feature matrix $N \times F$ feature matrix, X, where N is the number of nodes and F is the number of input features for each node, and
- an $N \times N$ matrix representation of the graph structure, such as the adjacency matrix A of G.

Figure 3 visually depicts the spatial distribution of the four monitoring stations and the proposed model architecture. The figure highlights how data from these stations are used to predict air quality at four different locations within

the network topology and illustrates the model's capability to capture temporal dynamics. The model is lightweight, featuring one GCN layer with 128 units and two LSTM layers with 128 units each.

3.3 Temporal Model

To capture temporal dependencies, we employ an optimized LSTM model, termed *Temporal LSTM* (T-LSTM), as detailed in our previous work [28]. This lightweight and compact model has proven effective for various nodes. The comprehensive framework of PMForecast designed for air pollution prediction is outlined, comprising four key steps: data pre-processing, temporal attention to mitigate gradient disappearance, a flexible prediction horizon for dynamic future forecasting, and layers employing Long Short-Term Memory (LSTM)—the trainable component. The temporal attention prediction horizon mechanisms are encapsulated in the block *Temporal Dynamic Updating* in Figure 3. This block is responsible for using temporal features such as the day of the week and also adjusting the prediction horizon for longer future forecasts. The LSTM model captures the temporal variations in the data, while the GCN model takes into account the topological structure and dependencies of the nodes.

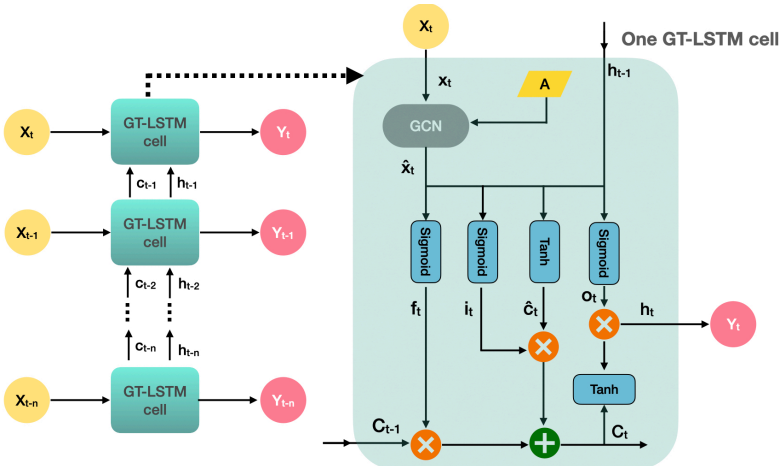

Fig. 4: A Cell of GT-LSTM

Figure 4 illustrates the architecture of a single GT-LSTM cell. The computational process within this cell is outlined by Equations 3 to 9. The GCN component, represented by Equation 3, processes input features x through the adjacency matrix A to generate spatial embeddings $\hat{x_t}$. The LSTM component, defined by Equations 4 to 9, captures temporal dependencies through the calculation of different gates:

$$\hat{x_t} = \sigma(W_g \cdot x_t \cdot A + b_g) \tag{3}$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, \hat{x_t}] + b_f) \tag{4}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, \hat{x_t}] + b_i) \tag{5}$$

$$\tilde{c_t} = \tanh(W_c \cdot [h_{t-1}, \hat{x_t}] + b_c) \tag{6}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c_t} \tag{7}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, \hat{x_t}] + b_o) \tag{8}$$

$$h_t = o_t \odot \tanh(c_t) \tag{9}$$

- $\hat{x_t}$ (GCN-transformed features): Updates node features by aggregating neighboring information via the normalized adjacency matrix $\tilde{A}$, and applying a non-linear activation σ.
- f_t (forget gate): Decides whether the information can pass through different layers of the network. It takes input from the previous hidden state h_{t-1} and the current input $\hat{x_t}$.
- i_t (input gate): Determines the importance of the information by updating the cell state. It measures the integrity and importance of the information for developing predictions. The information passes through the sigmoid and tanh functions; the tanh eliminates the bias of the network, and the sigmoid determines the weight of the information.
- $\tilde{c_t}$ (cell state candidate): Represents the new candidate values that could be added to the cell state.
- c_t (cell state): The cell state is updated by combining the forget gate and input gate outputs.
- o_t (output gate): The correct information passes through the cell state. Once here, the output of the input gate and forget gate is multiplied by each other. The output gate determines the next hidden state of the network.
- h_t (hidden state): The output gate decides the next hidden state. The updated cell state c_t goes through the tanh function and is multiplied by the sigmoid function of the output state.

The weighted parameters W in these equations are learned during the training process. The final output, h_t, represents the predicted air pollution level for the current time step at a specific node.

4 Sensor Data Description & Pre-processing

In our study, we employed data collected from advanced air quality micro-stations, called 'QameleO', developed by the University of Burgundy [21]. QameleO is an affordable micro-station for monitoring air quality, developed collaboratively by two research teams from the University of Burgundy and the *Institut de Recherches pour le Développement* (IRD). The QameleO network was deployed to augment the existing air quality reference network operated by ATMO

Bourgogne Franche Comté (ATMO-BFC) in the Dijon Metropolis. This reference network has been managed by regional governments since the 1990s. These micro-stations were deployed as part of the POPSU (*Plateforme d'Observation et de Stratégies Urbaines*) program within the Metropolis framework. Four locations (Port du Canal, Hoche, Carnot, and Janin) were selected to encompass various urban conditions, encompassing differing levels of traffic, background pollution, and intermediary scenarios. The dataset covered a period of one year, spanning from November 2020 to October 2021.

The data utilized in our analysis was collected at 15-minute intervals and encompassed five distinct measurements, including levels of particulate matters (PM_{10}, $PM_{2.5}$, PM_1), alongside meteorological variables : temperature and relative humidity. The QameleO dataset underwent comprehensive cleaning using advanced statistical algorithms at the University of Burgundy.

4.1 Data Acquisition and Preliminary Processing

The QameleO micro-station underwent thorough testing, encompassing both laboratory and outdoor evaluations, as part of a national air quality monitoring initiative led by INERIS (*Institut National de l'Environnement Industriel et des Risques*) and IMT (*Institut Mines Télécom*) Lille Douai, under the LCSQA (*Laboratoire Central de Surveillance de la Qualité de l'Air*). These assessments affirmed that the QameleO micro-station accurately captures the temporal variations in PMs mass concentrations. Measurements from QameleO micro-stations are captured every minute and then aggregated into 15-minute intervals to synchronize with the time-step of the ATMO reference stations deployed in Dijon Metropolis.

The data sets underwent various preprocessing stages as described in our previous work [28]. These stages included converting data from quarterly to hourly intervals. Then, we applied a moving average to effectively manage missing data and improve overall data quality. We utilized a sliding window technique with a 12-hour interval for each data point. Time-related features, such as day of the week and hour of the day, were integrated into the datasets, followed by normalization of values between zero and one. For a comprehensive understanding of this preprocessing pipeline, please refer to our prior study [28].

5 Experimental Results and Discussion

5.1 Model Performance Evaluation and Analysis

To evaluate the model's performance, we employed a combination of visual and quantitative analysis.

We visualized the correlation between observed and predicted values for the test sets, using density plots smoothed with a Gaussian kernel function (cf. Figure 5). Solid and dashed lines represent the true and predicted values, respectively, with colors corresponding to the four monitoring sites: Canal (red),

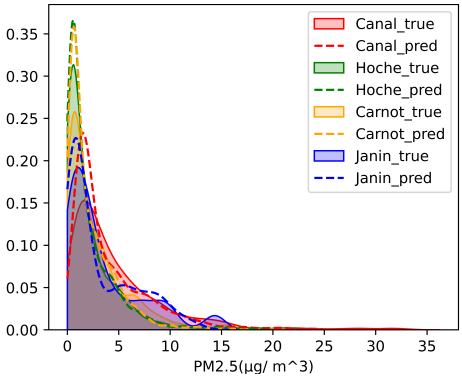

Fig. 5: Examination of the correlation between observed and predicted values for the test sets, smoothed using a Gaussian kernel function. The ground truth and predicted values are shown with solid and dashed lines, respectively. Colors represent the four sites in our datasets: Canal (red), Hoche (green), Carnot (orange), and Janin (purple).

Hoche (green), Carnot (orange), and Janin (purple). The close alignment between the curves demonstrates the model's strong predictive accuracy and consistency across all locations. The Gaussian distribution curve further emphasizes the model's accuracy, with its peak closely aligning with the predicted values, as represented by the dashed lines in the graph.

To further evaluate the model's ability to capture spatial dependencies, we conducted a detailed analysis focusing on two closely located sites (2 kilometers apart) : Hoche and Canal. The model effectively identifies similar air pollution patterns at these locations and produces consistent forecasting behaviors for both sites as it is shown in Figure 6.

5.2 Evaluating the Model's Ability to Capture Spatiotemporal Patterns

One of the primary objectives of this study is to generate predicted values for various nodes by inputting different datasets into the model, with a particular focus on the $PM_{2.5}$. We provide compelling evidence that leveraging spatiotemporal modeling, which considers dependencies between nodes, enables accurate forecasting of pollution levels. This assertion is supported by results from four monitoring sites and notably Node 4 at the *Janin*. Indeed, despite unavailability of measurements at that site for a certain period in the test set, our model predicted the $PM_{2.5}$ values for this site based on other site predictions and dependencies between all of them. Figure 7 illustrates the test-set data for all nodes, with the *Janin* site highlighted in the lower-right section of the graph. This achievement demonstrates the model's ability to capture spatiotemporal relationships and deliver reliable forecasts, even when actual measurements are

missing. It also reflects the model's consistent performance across different air pollutants.

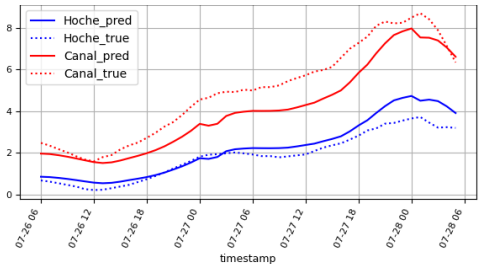

Fig. 6: Observed and predicted values for the Canal and Hoche sites over two days. Dotted lines represent the true values, while solid lines indicate the predicted values. The Hoche data is shown in blue, and the Canal data is shown in red.

A additional test was conducted to assess the stability of the model by artificially set Carnot site data to zero for the test set and predicting $PM_{2.5}$ values for all four nodes. Figure 8 illustrates both scenarios for the test set at the Carnot site: with real measurements and inputs set at zero. For Carnot, the predicted $PM_{2.5}$ values in both scenarios are similar and captured the ground truth data patterns well. However, the results show that in the scenario without real measurements (called zero), the range of predicted $PM_{2.5}$ values is narrower in comparison to the classic scenario with real inputs (see Figure 8 on the right). This indicates the model's limitation in distinguishing outliers when no actual input is provided, which is expected since the model lacks the historical data to predict values accurately. For the three other sites, there was a slight increase in error.

It is important to note that, in contrast, the temporal model [28] is unable to predict meaningful values when fed with zero or corrupted measurements. Instead, it either predicts zero or simply mirrors the flawed input data, relying exclusively on the data from the single station.

5.3 Assessing Model Capability for Long-Horizon Forecasting

Our forecasting model strongly predicts air pollution levels over extended time spans. Table 1 presents error and correlation metrics for test set across various time intervals, ranging from 1 hour to 36 hours into the future. These values represent the average of $PM_{2.5}$ predictions across all nodes, aggregated for each specific forecast horizon.

In Table 1, the model shows notable error reduction within the training dataset, achieving an average *Root Mean Square Error* (RMSE) value of 2.195 for the first 12 hours in the test set. This indicates a significant correlation between

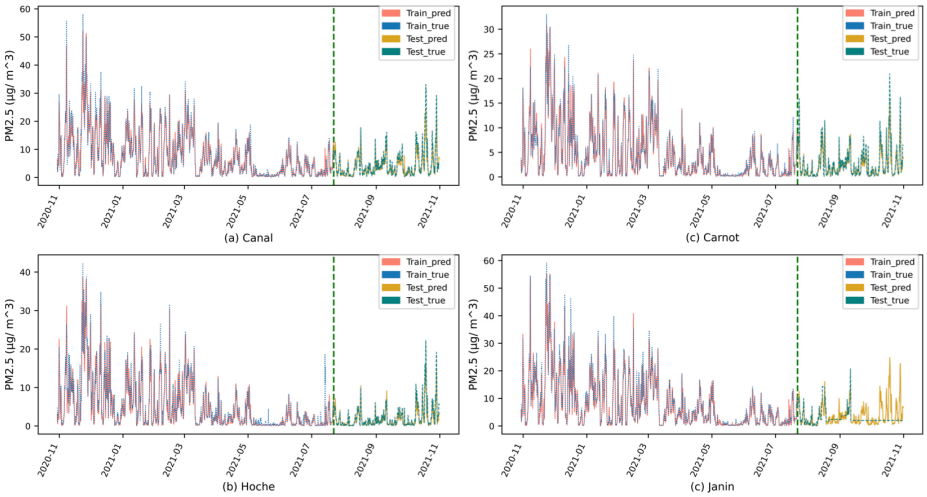

Fig. 7: Illustrating Model Robustness: predictions for all locations. The dashed lines represent the collected data, reflecting the actual values during both the training (blue) and prediction (golden) phases. The solid lines depict the $PM_{2.5}$ predictions made during the training (salmon) and prediction (green) phases. The vertical green dashed line marks the boundary between the training and testing datasets.

Table 1: Evaluation Metrics for 1 to 36 Hours Forecasting

Hour(s)	RMSE	MAE	MSE	WMAPE (%)	R^2
1 Hour	1.433	0.786	2.138	18.3	0.892
6 Hours	1.702	1.001	3.103	28.7	0.811
12 Hours	2.195	1.277	5.360	35.2	0.643
24 Hours	2.635	1.429	8.197	47.5	0.458
36 Hours	3.112	1.712	10.569	59.1	0.356

predicted and actual values. The model also exhibits robust performance on the independent test dataset, with an average *Mean Absolute Error* (MAE) value of 1.277, demonstrating its ability to generalize effectively and provide reliable predictions even for unforeseen data beyond the 12-hour mark.

Table 1 evaluates the model's accuracy over longer forecast horizons. After 24 hours, the accuracy of predictions decreases by more than 50 percent. Despite this, the model's performance indicates that extending forecasts beyond this period presents challenges. To improve long-term prediction quality, especially with large datasets, it is recommended to consider daily average predictions after the 24-hour mark.

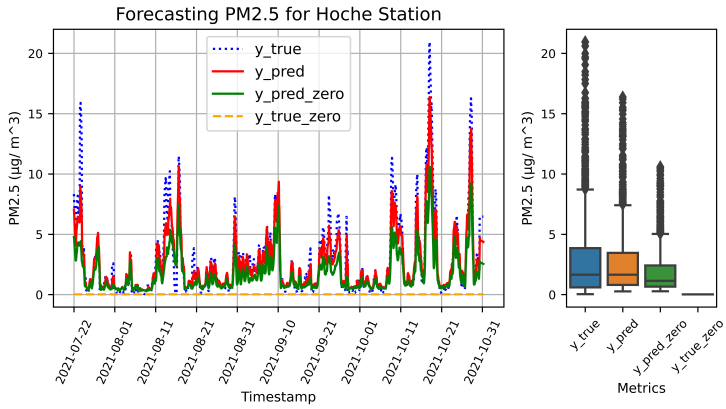

Fig. 8: Assessing Model Robustness: Two testing scenarios are considered. (i) Without Real Measurements (Using Zero Values): The dashed orange line represents zero values as input, while the solid green line shows the predicted values for this scenario. (ii) With Real Ground-Truth Measurements: The dotted blue line represents the actual ground-truth values, and the solid red line depicts the $PM_{2.5}$ predictions made using the real measurements for this scenario.

5.4 Comparison of Local and Network-Based Forecasting Models

In this section, we analyze and compare key characteristics of the temporal model (PMForecast) [28] and the proposed spatiotemporal model GT-LSTM. PMForecast, which has shown strong predictive performance against baselines such as GRU and XGBoost, serves as the backbone for the temporal component of GT-LSTM. Building on this foundation, GT-LSTM incorporates spatial dependencies to enhance forecasting across multiple locations.

As detailed in Table 2, we evaluate both models across four dimensions: performance, computational efficiency, scalability, and data privacy. PMForecast demonstrates very high predictive accuracy (MAE: 0.164, Accuracy: 98%) but is limited to forecasting at a single station, with notable degradation when scaling to multiple sites. In contrast, GT-LSTM achieves high overall performance (MAE: 0.786, Accuracy: 89%) while offering strong scalability—maintaining prediction quality across multiple monitoring stations without significant loss in accuracy. Although both models exhibit low computational efficiency, GT-LSTM is more efficient in terms of broader coverage, training on four stations in the same time PMForecast trains on one.

With respect to data privacy, both models rely on basic handling without encryption, highlighting an important limitation for deployment in sensitive or

regulated environments. Future work could address this gap through privacy-preserving approaches such as data anonymization or federated learning.

Table 2: Comparison of Temporal and Spatio-Temporal Forecasting Models

Dimension	**PMForecast (Temporal Model)**	**GT-LSTM (Spatio-Temporal Model)**
Performance	Very High — MAE: 0.164, Accuracy: 98%	High — MAE: 0.786, Accuracy: 89%
Computational Efficiency	Low — Train Full Model Time: 212s for 1 station	Low — Train Full Model Time: 423s for 4 stations
Scalability	Low — Limited to 1 station	High — Scales well to multiple stations
Data Privacy	Low — No encryption, basic data handling	Low — No encryption, basic data handling

5.5 Experimental Setup

The performance of deep learning models is highly influenced by hyperparameters such as learning rate, batch size, number of epochs, and hidden layer size. In this study, we performed limited but targeted hyperparameter tuning to identify configurations that provide stable performance without overfitting.

For the GCN component, we evaluated hidden unit sizes of $[32, 64, 128, 256]$, with 128 units yielding the most consistent results. Among tested configurations, a single-layer GCN with Sigmoid activation provided the best trade-off between performance and complexity. This setup was effective in capturing local spatial relationships based on the available sensor data.

The temporal component followed the structure used in our prior work, which we found to be adequate for this dataset. It consists of two stacked LSTM layers, each with 128 units, using ReLU activation. This configuration offered a good balance between model capacity and computational efficiency.

We trained the model using the Huber loss function, selected for its robustness to outliers while maintaining precision for small errors. The learning rate was set to 0.001, and training was performed over 200 epochs with early stopping based on validation loss to reduce overfitting risk.

Although the experimental setup is limited by the number of available sensors—potentially restricting generalization and spatial diversity—the chosen configuration aimed to ensure fair evaluation under data-constrained conditions. Future work will explore scaling to denser sensor networks and broader spatial settings.

6 Conclusion

In conclusion, this study presents a spatiotemporal model designed to predict $PM_{2.5}$ concentrations in urban and suburbans environments. This work extends our previously validated temporal model [28] by integrating spatial learning through GCN, resulting in improved accuracy under sparse sensor conditions. Leveraging a low-cost sensor network and advanced AI techniques, the model effectively captures spatial and temporal air pollution patterns while maintaining a lightweight and scalable architecture.

Beyond predictive accuracy, the model offers practical value for long-term planning and targeted forecasting in subregions with limited data. Its real-time monitoring capability supports policymakers and local communities, enhancing air quality management and contributing to better public health outcomes.

7 Future Work

As part of future work, we propose integrating mobile sensor networks to improve the spatiotemporal resolution of air pollution monitoring. Deploying mobile sensors across urban areas will facilitate the collection of high-resolution, localized air quality data, enabling more precise and adaptive forecasting. To preserve data privacy while supporting distributed model training, we also plan to incorporate federated learning into the framework.

8 Acknowledgement

This project is funded by the European Union's Horizon 2020 research and innovation program (Marie Sklodowska-Curie agreement No. 847568). The research is conducted with the *Centre de Recherche en Informatique, Signal et Automatique de Lille* (CRIStAL) in partnership with Inria Lille and the *Laboratoire d'Optique Atmosphérique* (LOA) at the University of Lille. Quality-checked QAMELEO data is provided by the University of Bourgogne under the RESPONSE H2020 program. The QAMELEO network in Dijon is funded by the Ministry of Ecological Transition, Territorial Cohesion, and Dijon Metropolis (POPSU PURE program) and installed in collaboration with the University of Bourgogne and *Institut de Recherches pour le Développement* (IRD), co-owners of the instruments (Creative Commons open source license since 2020). The implementation of the Graph Temporal LSTM model used in this work are publicly available at [27].

References

1. Alexander Baklanov, Y.Z.: Advances in air quality modeling and forecasting. Global Transitions **2**, 261–270 (2020). https://doi.org/doi.org/10.1016/j.glt.2020.11.001

2. Belavadi, S.V., Rajagopal, S., R, R., Mohan, R.: Air quality forecasting using lstm rnn and wireless sensor networks. Procedia Computer Science **170**, 241–248 (2020). https://doi.org/10.1016/j.procs.2020.03.036, https://www.sciencedirect.com/science/article/pii/S1877050920304658
3. Deakin, E.: Sustainable development and sustainable transportation: Strategies for economic prosperity, environmental quality, and equity. Tech. rep., UC Berkeley: Institute of Urban and Regional Development (2001), https://escholarship.org/uc/item/0m1047xc
4. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. CoRR **abs/1606.09375** (2016), http://arxiv.org/abs/1606.09375
5. Drewil, G.I., Al-Bahadili, R.J.: Air pollution prediction using lstm deep learning and metaheuristics algorithms. Measurement: Sensors **24**, 100546 (2022). https://doi.org/https://doi.org/10.1016/j.measen.2022.100546
6. García-Duarte, L., Cifuentes, J., Marulanda, G.: Short-term spatio-temporal forecasting of air temperatures using deep graph convolutional neural networks. Stochastic Environmental Research and Risk Assessment **37**(5), 1649–1667 (2023). https://doi.org/https://doi.org/10.1007/s00477-022-02358-0
7. Ge, L., Wu, K., Zeng, Y., Chang, F., Wang, Y., Li, S.: Multi-scale spatiotemporal graph convolution network for air quality prediction. Applied Intelligence **51**(6), 3491–3505 (2021). https://doi.org/10.1007/s10489-020-02054-y, https://doi.org/10.1007/s10489-020-02054-y
8. Ge, L., Wu, K., Zeng, Y., Chang, F., Wang, Y., Li, S.: Multi-scale spatiotemporal graph convolution network for air quality prediction. Appl. Intell. **51**(6), 3491–3505 (2021). https://doi.org/10.1007/S10489-020-02054-Y, https://doi.org/10.1007/s10489-020-02054-y
9. Gilik, A., Ogrenci, A.S., Ozmen, A.: Air quality prediction using cnn+lstm-based hybrid deep learning architecture. Environmental Science and Pollution Research **29**(8), 11920–11938 (2022). https://doi.org/10.1007/s11356-021-16227-w, https://doi.org/10.1007/s11356-021-16227-w
10. Gokul, P., Mathew, A., Bhosale, A., Nair, A.T.: Spatio-temporal air quality analysis and pm2.5 prediction over hyderabad city, india using artificial intelligence techniques. Ecological Informatics **76**, 102067 (2023). https://doi.org/https://doi.org/10.1016/j.ecoinf.2023.102067, https://www.sciencedirect.com/science/article/pii/S1574954123000961
11. Guo, Z., Jing, X., Ling, Y., Yang, Y., Jing, N., Yuan, R., Liu, Y.: Optimized air quality management based on air quality index prediction and air pollutants identification in representative cities in china. Scientific Reports **14**(1), 17923 (2024). https://doi.org/10.1038/s41598-024-68972-w, https://doi.org/10.1038/s41598-024-68972-w
12. He, Y., Ma, J., Zhang, C., Yang, H.: Spatio-temporal evolution and prediction of carbon storage in guilin based on FLUS and InVEST models. Remote Sensing **15**, 1445. https://doi.org/10.3390/rs15051445
13. Hofman, J., Do, T.H., Qin, X., Bonet, E.R., Philips, W., Deligiannis, N., La Manna, V.P.: Spatiotemporal air quality inference of low-cost sensor data: Evidence from multiple sensor testbeds. Environmental Modelling & Software **149**, 105306 (2022). https://doi.org/https://doi.org/10.1016/j.envsoft.2022.105306, https://www.sciencedirect.com/science/article/pii/S1364815222000123
14. Jain, A., Bhasin, A., Gupta, V.: Prediction of air pollution using lstm-based recurrent neural networks. Int. J. Comput. Intell. Stud. **8**(4), 299–

308 (2019). https://doi.org/10.1504/IJCISTUDIES.2019.103620, https://doi.org/10.1504/IJCISTUDIES.2019.103620
15. Kim, K.H., Kabir, E., Kabir, S.: A review on the human health impact of airborne particulate matter. Environment International **74**, 136–143 (2015). https://doi.org/https://doi.org/10.1016/j.envint.2014.10.005, https://www.sciencedirect.com/science/article/pii/S0160412014002992
16. Krzyzanowski, M., Apte, J.S., Bonjour, S.P., Brauer, M., Cohen, A.J., Prüss-Ustun, A.M.: Air pollution in the mega-cities. Tech. Rep. 3, Current Environmental Health Reports (2001). https://doi.org/10.1007/s40572-014-0019-7
17. Le, V.D.: Spatiotemporal graph convolutional recurrent neural network model for citywide air pollution forecasting (2023), https://arxiv.org/abs/2304.12630
18. Le, V.D., Bui, T.C., Cha, S.K.: 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 (2020). https://doi.org/10.1109/BigComp48618.2020.00-99
19. Lee, H.M., Park, R.J., Henze, D.K., Lee, S., Shim, C., Shin, H.J., Moon, K.J., Woo, J.H.: Pm2.5 source attribution for seoul in may from 2009 to 2013 using geos-chem and its adjoint model. Environmental Pollution **221**, 377–384 (2017). https://doi.org/https://doi.org/10.1016/j.envpol.2016.11.088
20. M, D., V, R.: Novel regression and least square support vector machine learning technique for air pollution forecasting. CoRR **abs/2306.07301** (2023). https://doi.org/10.48550/ARXIV.2306.07301, https://doi.org/10.48550/arXiv.2306.07301
21. Martiny, N., Nicolas, M., Sarah, M., Julita, D.D., Canovas, L., Bisquerra, A., Rega, M., Thévenin, T.: Quality of air module for environmental learning engineering and observation network (qameleondijon) : un réseau dense de mesures de qualité de l'air à dijon», climatologie **20**(4) (2023). https://doi.org/https://doi.org/10.1051/climat/202320004
22. Norby, R.J., Luo, Y.: Evaluating ecosystem responses to rising atmospheric co2 and global warming in a multi-factor world. New Phytologist **162**(2), 281–293 (2004). https://doi.org/https://doi.org/10.1111/j.1469-8137.2004.01047.x
23. Organization, W.H.: Air pollution and child health. World Health Organization (2006)
24. Pak, U., Ma, J., Ryu, U., Ryom, K., Juhyok, U., Pak, K., Pak, C.: Deep learning-based pm2.5 prediction considering the spatiotemporal correlations: A case study of beijing, china. Science of The Total Environment **699**, 133561 (2020). https://doi.org/https://doi.org/10.1016/j.scitotenv.2019.07.367, https://www.sciencedirect.com/science/article/pii/S0048969719334813
25. Pakrooh, P., Pishbahar, E.: Forecasting air pollution concentrations in iran, using a hybrid model. Pollution **5**(4), 739–747 (2019). https://doi.org/10.22059/poll.2019.274827.572
26. Qin, D., Yu, J., Zou, G., Yong, R., Zhao, Q., Zhang, B.: A novel combined prediction scheme based on cnn and lstm for urban pm2.5 concentration. IEEE Access **7**, 20050–20059 (2019). https://doi.org/10.1109/ACCESS.2019.2897028
27. Rahmani, M.: Graph temporal lstm for air quality forecasting. https://github.com/Maryamr92/GT-LSTM (2024), accessed: 2025-04-18
28. Rahmani, M., Crumeyrolle, S., Allegri-Martiny, N., Taherkordi, A., Rouvoy, R.: Pmforecast: leveraging temporal lstm to deliver in situ air quality predictions. Environmental Science and Pollution Research (2024). https://doi.org/10.1007/s11356-024-34623-w, https://doi.org/10.1007/s11356-024-34623-w

29. Shakir, M., Kumaran, U., Rakesh, N.: An approach towards forecasting time series air pollution data using lstm-based auto-encoders. J. Internet Serv. Inf. Secur. **14**(2), 32–46 (2024). https://doi.org/10.58346/JISIS.2024.I2.003, https://doi.org/10.58346/jisis.2024.i2.003
30. Yin, L., Liu, P., Wu, Y., Shi, C., Wei, X., He, Y.: ST-VGBiGRU: A hybrid model for traffic flow prediction with spatio-temporal multimodality. IEEE Access **11**, 54968–54985. https://doi.org/10.1109/ACCESS.2023.3282323
31. Zhang, Q., Han, Y., Li, V.O.K., Lam, J.C.K.: Deep-air: A hybrid cnn-lstm framework for fine-grained air pollution estimation and forecast in metropolitan cities. IEEE Access **10**, 55818–55841 (2022). https://doi.org/10.1109/ACCESS.2022.3174853