Nationwide Air Pollution Forecasting with Heterogeneous Graph Neural Networks

FERNANDO TERROSO-SAENZ, Catholic University of Murcia (UCAM), Spain
JUAN MORALES-GARCÍA, Catholic University of Murcia (UCAM), Spain
ANDRES MUÑOZ, University of Cádiz, Spain

Nowadays, air pollution is one of the most relevant environmental problems in most urban settings. Due to the utility in operational terms of anticipating certain pollution levels, several predictors based on Graph Neural Networks (GNN) have been proposed for the last years. Most of these solutions usually encode the relationships among stations in terms of their spatial distance, but they fail when it comes to capture other spatial and feature-based contextual factors. Besides, they assume a *homogeneous* setting where all the stations are able to capture the same pollutants. However, large-scale settings frequently comprise different types of stations, each one with different measurement capabilities. For that reason, the present paper introduces a novel GNN framework able to capture the similarities among stations related to the land use of their locations and their primary source of pollution. Furthermore, we define a methodology to deal with heterogeneous settings on the top of the GNN architecture. Finally, the proposal has been tested with a nation-wide Spanish air-pollution dataset with very promising results.

CCS Concepts: • **Computing methodologies** → **Machine learning algorithms**; **Artificial intelligence**; • **Information systems** → **Information systems applications**.

Additional Key Words and Phrases: air pollution, graph neural networks, forecasting, nationwide scale

ACM Reference Format:
Fernando Terroso-Saenz, Juan Morales-García, and Andres Muñoz. 2023. Nationwide Air Pollution Forecasting with Heterogeneous Graph Neural Networks. In . ACM, New York, NY, USA, 20 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

One of the most relevant environmental problems in our modern society is the poor air quality in numerous urban areas. In fact, the World Health Organization (WHO) remarks that 99% of the worldwide population are in settlements where air pollution levels exceed the WHO guideline limits[1]. This same organization has highlighted that the exposure to air pollutants causes a higher mortality rate in different regions [28].

In this context, public authorities have promoted the deployment of sensors to measure multiple pollutants produced by road-traffic activities (NO_2, CO, NH_3, PM2.5 or PM10) or industry sources (SO_2, CO or O_3) [2, 3] in real time. Thanks to this new source of data, the research community have proposed a large number of forecasting solutions to anticipate the level of pollutants in a specific area at multiple time horizons [6, 11, 26, 27]. In this manner, institutions and individuals could be able to take proactive actions before a critical pollution level is reached [4].

[1]https://www.who.int/data/gho/data/themes/air-pollution/ambient-air-pollution

Manuscript submitted to ACM

Regarding the applied methods for such a forecasting task, multiple algorithms dealing with timeseries have been tested, such as ARIMA [26], Support Vector Machines [17] or deep learning techniques like recurrent [15] or convolutional [9] neural networks. Due to the spatial and temporal dependencies involved in the prediction of air pollution, several approaches have recently proposed the use of Graph Neural Networks (GNNs) to encode such spatiotemporal relationships as part of the inference process [7, 10, 13, 14]. Despite the promising results obtained by the GNN approach, we can still identify two major limitations in the current literature:

- On the one hand, GNN solutions usually consider the distance among sensors as the only relevant spatial feature. However, there are other spatial factors that might be relevant in the air-pollution forecasting pipeline. For example, the land use or the prominent human activity in the sensor's surrounding area. Thus, the pollution patterns might be different whether a sensor is installed in a urban, rural or suburban area or whether it is installed closed to a motorway or an industrial park.
- On the other hand, most solutions assume that the infrastructure of the monitoring system is homogeneous and all the sensors are able the measure the same pollutants. However, most real air pollution measurement networks deployed at large scale are very heterogeneous. In fact, they generally comprise different types of sensors, each one able to measure a different palette of pollutants. As a result, there are very few proposals that actually consider this factor as part of their forecasting method.

Our work proposes a forecasting method to anticipate several air pollutants at large spatial scales that overcomes these two drawbacks. To do so, the proposed methodology encodes the similarity among the air measurement stations in terms of their surrounding land usage and their primary source of pollution. Our model relies on a GNN architecture that receives as input an heterogeneous graph encoding these similarities along with other contextual factors such as human mobility activity and weather conditions. Moreover, the stations are heterogeneous, containing several types of sensors measuring different types of pollutants. Consequently, our proposal also considers such heterogeneity to better represent real-world scenarios. In that sense, it is possible to find in the literature different techniques to deal with heterogeneous graphs and generate the node embeddings under different conditions [24]. Moreover, they have been successfully used in different settings such as the development of recommendation systems in the e-commerce field [8]. The solution has been tested against a nation-wide air-pollution dataset in Spain comprising more than 600 air measurement stations.

The remainder of the paper is structured as follows. Section 2 gives an overview about existing trends and techniques for air pollution forecasting. Then, section 3 introduces the methodology to define the GNN architecture as a pollutant predictor. Section 4 states the evaluation of the proposed predictor. Lastly, section 5 summarizes the main conclusions and potential future research lines motivated by this work.

2 RELATED WORK

The methods for analyzing and predicting air pollution have experienced significant advances, specially thanks to the recent explosion of numerous Machine Learning (ML) and Deep Learning (DL) models. By leveraging large-scale datasets, it is now possible to develop highly accurate predictive models that capture the relationships between meteorological conditions, emissions sources, and pollutant levels.

Starting with related papers using ML methods, in [26] the ARIMA model was applied to analyze the effect of land surface coverage (LSC) on PM_{10} pollutants in Bogotá, Colombia. The data gathered by 6 monitoring stations during 6 years included PM_{10} concentration, temperature, solar radiation and wind speed and direction. Using these data, the

authors studied the relationship between PM_{10} concentrations and different LSC areas in a daily basis to simulate the spread of this pollutant. The findings from the study revealed that areas with greater vegetation maintained higher levels of PM_{10}, preventing its dispersion into surrounding areas. Thus, a difference of about 42% of PM_{10} concentration was observed in favor of green areas compared to areas with predominance of pavements and buildings. In addition, the work in [17] proposed the use of the support vector machine (SVM) algorithm to predict the Air Quality Index (AQI) for the city of Ahmedabad, India. This AIQ is calculated using 12 pollutant indices, including PM_{10}, $PM_{2.5}$ and CO, among others. The authors leveraged the data collected by 7 air monitoring stations in the city monitoring the 12 pollutants during 6 years. The proposed SVM algorithm trained with this dataset obtained a R^2 of 0.998, outperforming other models compared during the study, such as LSTM and SARIMA. However, no information was given about the time horizon of the prediction.

Regarding works leveraging DL techniques, an excellent survey on deep learning neural networks (DNN) can be found elsewhere [27]. This survey states that these models are able to outperform previous models of ML such as artificial neural networks when large volumes of historic air pollution data are available. In particular, when there is a strong variation in these dataset, Recurrent Neural Networks (RNN) and Convolutional Neuronal Networks (CNN) are the preferred models. Thus, the paper in [15] offers an example of an RNN for predicting PM_{10} and SO_2. The authors collected data about these pollutants from the industrialized province of Sakarya (Turkey) over a two-year period using an open data initiative by the Ministry of Environment of that country. The results showed that the proposed RNN was able to predict better the SO_2 values compared to the PM_{10} ones, with an RMSE difference of one order of magnitude. Again, no information was given about the time horizon of the prediction. Likewise, the work in [9] used a CNN for hourly predictions of PM_{10}, SO_2 and O_3 in three different cities, namely Barcelona, Kocaeli and Istanbul. Despite CNN usually being applied to image analysis, the authors adapted this technique to analyze 2D and 3D spatial matrices representing the historical data on a specific pollutant and its relationship to other pollutants. They combined the CNN with an LSTM layer to analyze the temporal features of the pollutants. They obtained RMSE values under 0.1 for all the pollutants in the cities of Kocaeli and Istanbul. Moreover, meteorological features were added to the model, but did not result in significant improvements. A transfer learning method was also applied by training the model on Kocaeli data and using it to predict pollution in Istanbul, yielding similar results to the baseline experiment.

Recent approaches in this area are exploring the performance of hybrid models combining ML/DL techniques. Thus, the authors in [6] proposed a decomposition/reconstruction model combining an adaptive noise algorithm with a K-means clustering method (CEEMDAN-KMC-RLN) to forecast the $PM_{2.5}$, PM_{10}, NO_2 and O_3 pollutants both at a specific time points and intervals. This model was trained and validated using a 6-year dataset on these pollutants. The results showed that the hybrid model outperformed the individual methods used in the proposed hybrid model. Another study [11] combined LSTM with multi-verse optimization metaheuristic algorithm to forecast NO_2 and SO_2. Thanks to this combination, the authors were able to optimize the LSTM parameters in an automatic manner through a mutual information method. This proposal was evaluated by predicting the pollutants in a power plant in Kerman, Iran. Apart from five months of pollutant data, the authors also used air temperature and wind speed as additional training features. The results demonstrated that this hybrid model achieved a better accuracy compared with other variations of LSTM and Elman neural network hybrid models.

A current trend that has attracted significant attention in numerous studies is the utilization of Graph Neural Networks (GNNs) for air quality forecasting. An example is found in [14], where the authors use a neural network known as Deep Spatio-Temporal Graph Network (BGGRU) to forecast the $PM_{2.5}$ particle in 16 districts of the region of Beijing (China). In addition, these studies usually use exogenous variables in order to improve the prediction quality.

Some examples of this are the study in [7], in which the authors use a neural network known as Dynamic Graph Convolutional network and the Multi-channel Temporal Convolutional Network (DGC-MTCN) to forecast the $PM_{2.5}$ particle in two regions (Beijing and Fushun), with the aid of auxiliary variables such as other pollutants (like $PM_{2.5}$, SO_2, NO_2, CO and O_3) and weather conditions as exogenous variables. Also the study in [10], in which the authors use a neural network known as Self-Supervised Hierarchical Graph Neural Network (SSH-GNN) to forecast the AQI, $PM_{2.5}$, PM_{10}, O_3, NO_2, SO_2, and CO particles in the urban agglomerations of Beijing-Tianjin-Hebei, Pearl River Delta and Yangtze River Delta, with the aid of auxiliary variables such human mobility and weather conditions as exogenous variables. Finally, the work in [13] leverages a neural network known as Attention Temporal Graph Convolutional Network (A3T-GCN) to forecast the NO_2 particle in the region of Madrid (Spain), with the aid of auxiliary variables such human mobility and weather conditions as exogenous variables.

Table 1. Overview of different methods for air pollution forecasting based on their prediction algorithms and input data.

Ref.	Spatial scale	Target Pollutants	Context factors	Homogeneous sensors?	Land-use?	Source of pollution?	Method
[26]	City	PM10	temperature, relative humidity, air pressure, wind speed and direction	No	No	N/A	CNN+LSTM
[17]	City	AiQ (PM10, PM2.5, NO2, SO2, CO, O3, NH3, Pb, Ni, As, Benzo(a)pyrene, and Benzene)	-	Yes	No	N/A	SVM
[9]	City	PM10, SO2, O3	-	Yes	No	N/A	SVM
[11]	City	SO2, NO2	Air temperature, wind speed	Yes	Yes	Power Plan	LSTM-MVO
[15]	District	PM10 & SO2	-	N/A	No	Industrial	RNN
[7]	Region	PM2.5	Other pollutants (PM10, SO2, NO2, CO, O3), weather conditions	No	No	No	DGC-MTCN
[10]	Urban agglomerations	AiQ (AQI, PM2.5, PM10, O3, NO2, SO2, and CO)	Human mobility, weather conditions	No	Yes	No	SSH-GNN
[13]	Region	NO2	Human mobility, weather conditions	No	Yes	Yes	A3T-GCN
[14]	Region	PM2.5	-	Yes	Yes	No	BGGRU
[6]	N/A	PM2.5, PM10, NO2, O3	-	N/A	No	N/A	CEEMDAN-KMC-RLN
Our approach	Nation	C6H6, PM10, PM2.5, SO_2, O_3, CO, NO, NO_2, NOx	Human mobility, weather conditions	No	Yes	Yes	ConvLSTM + GAT

Table 1 summarizes the most important features of the reviewed works. As observed, only a limited number of studies actually consider non-homogeneous sensor settings [7, 10, 13]. However, their focus is primarily on predicting a single pollutant (PM2.5, NO_2) or an aggregated Air-Quality index (AIQ). On the contrary, our work tackles a multi-variate prediction by anticipating the values of 9 pollutants. In addition to that, our work also combines the information about the potential pollution sources and the type of urban area (urban, rural of suburban) where the station is located. While it is true that other models have utilized similar contextual data such as POIs distribution or road-network topology [10], our work defines and incorporates this contextual data in a more concise and simplified manner. This allows composing a lightweight model architecture that is less prone to overfitting, as demonstrated in sec. 3.2.

3 METHODOLOGY TO GENERATE THE GRAPH NEURAL NETWORKS

In this section, we describe the procedure to generate the GNN model by firstly describing the different datasets to compose the model and then its inner architecture.

3.1 Datasets

We have used 3 different datasets to compose the predictor. The main features of each one are stated next.

3.1.1 Air pollution Dataset. This work makes use of a public nationwide air pollution dataset released by the Spanish Ministry for Ecological Transition (MET)[2]. The dataset is obtained from 633 air monitoring stations located at different points of the Spanish territory, as Fig. 1a shows. Note that the two largest cities in Spain, Madrid and Barcelona, are the ones including the highest number of stations, as shown in Fig. 2. Moreover, 389 Spanish municipalities have at least one monitoring station. In particular, each of these cities have, on average, 1.63 with a standard deviation of ±1.90.

(a) Location of the stations of the air-pollution dataset.

(b) Location of the stations based on the land usage of their associated area.

(c) Location of the stations based on their primary source of pollution.

Fig. 1. Spatial distribution of the air monitoring stations in Spain .

The dataset comprises the levels of 9 pollutants on an hourly basis, namely nitrogen monoxide (NO), sulphur dioxide (SO_2), nitrogen dioxide and nitrogen oxides (NO2, NO_X), particulate matter (PM10 and PM2.5), benzene (C6H6), carbon monoxide (CO) and ozone (O3). Let us call this set of particle types as $\mathcal{P}$. However, not all the stations are able to measure such 9 particles. Fig. 3a shows the distribution of stations based on the number of pollutants that they are able

[2]https://www.miteco.gob.es/es/calidad-y-evaluacion-ambiental/temas/atmosfera-y-calidad-del-aire/calidad-del-aire/evaluacion-datos/datos/Datos_2001_2020.aspx

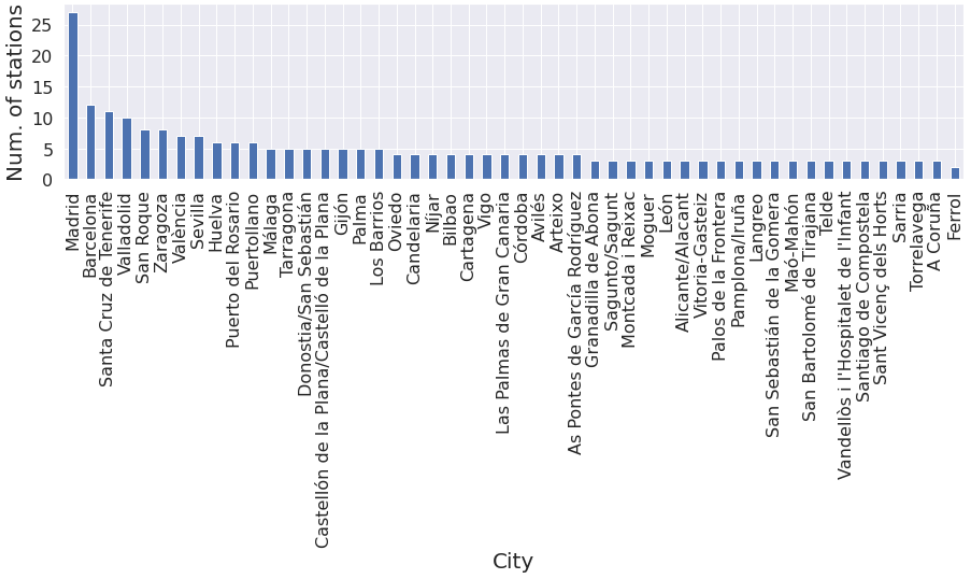


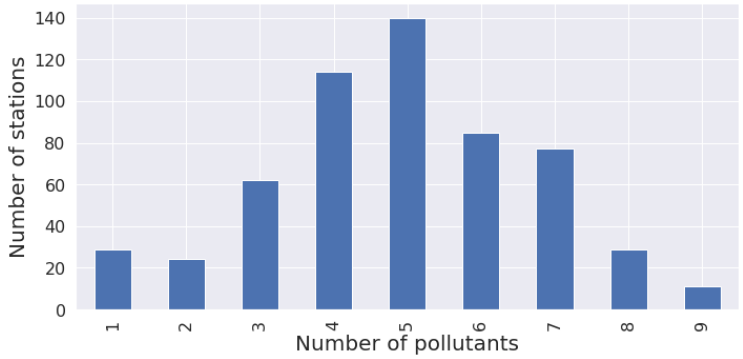
Fig. 2. Number of stations per Spanish municipality.

to measure. For example, it can be observed that only 11 stations are able to measure the 9 target pollutants whereas 140 measure only 5 of them. Besides, Fig. 3b shows the distribution of stations per particle, for example it is observed that 363 stations are able to measure ozone (O3) levels and only 49 of them NO particles. Hence, these two plots show the significant heterogeneity in the underlying air pollution sensor infrastructure that compose the dataset.

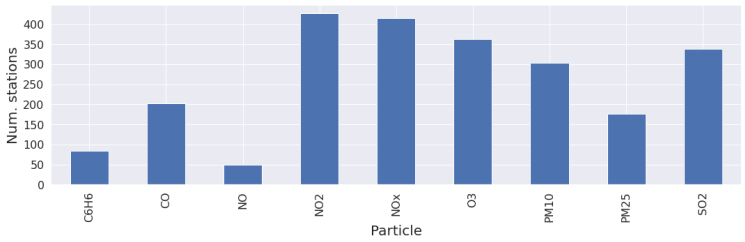
Furthermore, the MET dataset also labels the 633 stations with 2 different tags. The first one indicates the primary land use of the surrounding area where the station is located. Thus, this land-use tag distinguishes among *urban*, *suburban* and *rural* stations with a distribution of 283, 213 and 137 rural stations, respectively. The urban stations are the ones located at the center of the most important Spanish cities, the suburban ones are installed in the outskirts of a city and the rural stations are installed in small villages or towns. The spatial distribution of the stations based on this tag is shown in Fig. 1b. Observe that urban and suburban stations are usually located quite close to each other, while the rural stations are more spatially isolated. This is consistent with the nature of their associated tags.

The second tag informs about the primary source of pollution measured by the station. This *pollution label* takes 3 values: *traffic*, when the station mainly measures pollution generated by road-traffic activity; *industry*, when the station is located near industrial parks; and *mixed*, when the station measures pollution from a mixed number of sources. Fig. 1c shows the spatial distribution of the stations based on this label.

Finally, Fig. 4 shows the co-occurrence distribution of these two tags. Note that 106 urban stations have traffic activity as the primary pollution source. This is consistent with the fact that most of the center of the cities have usually a high density of vehicles. On the other hand, the most frequent pollution source of the suburban stations are mixed (100). Likewise, this makes sense as many industrial parks tend to be located on the outskirts of towns and cities. Finally, the 85 rural stations also have a mixed combinations of sources. This is also consistent with the fact that small towns typically have neither a significant volume of traffic nor large-scale industrial activity. As a result, it is reasonable to conclude that small towns do not possess a distinct primary source of pollution.



(a) Distribution of pollutants per station.



(b) Distribution of stations per pollutant.

Fig. 3. Relation of the distributions between pollutants and stations.

Consequently, if an air-pollution station $s \in \mathcal{S}$, where $\mathcal{S}$ is the set, is able to measure m different pollutants then it is possible to define a multivariate timeseries with n lags at hour h, $\mathcal{T}_h^s = \langle t_{h-n}^s \rightarrow t_{h-n+1}^s \rightarrow t_{h-n+2}^s \cdots \rightarrow t_h^s \rangle$ where a tuple $t_j^s = \langle p_{j,1}^s, p_{j,2}^s, .., p_{j,m}^s \rangle$ comprises the values of the m pollutants at the j-th hour.

3.1.2 Human mobility dataset. Due to the relevance of the road traffic activity as a primary factor that drives the air pollution patterns of a spatial region [18], we have also used a human mobility dataset in our study.

This dataset has been retrieved from the nationwide human mobility report released by the Spanish Ministry of Transportation (SMT) in December 2020[3]. It covers a 15-month period from February 29th, 2020 to May 10th, 2021, indicating the number of trips among 3216 ad-hoc administrative areas (hereby *Mobility Areas, MA*) per hour in Spain both in its peninsular and insular extension. A *single trip* stands for the spatial displacement of an individual with distance above 500 meters. Consequently, this dataset could be regarded as a set of tuples where each one takes the form

$$\langle date, hour, m_{origin}, m_{dest}, n_{trp}, dist \rangle$$

reporting that there is n_{trp} human trips from the MA m_{origin} to the MA m_{dest} covering a distance of $dist$ km during the indicated $date$ and $hour$. Fig. 5 shows the spatial boundaries of the MAs defined in the dataset as blue lines along with the location of the Spanish airports as red points which we will discuss in sec. 3.1.3.

[3]https://www.mitma.es/ministerio/covid-19/evolucion-movilidad-big-data/opendata-movilidad

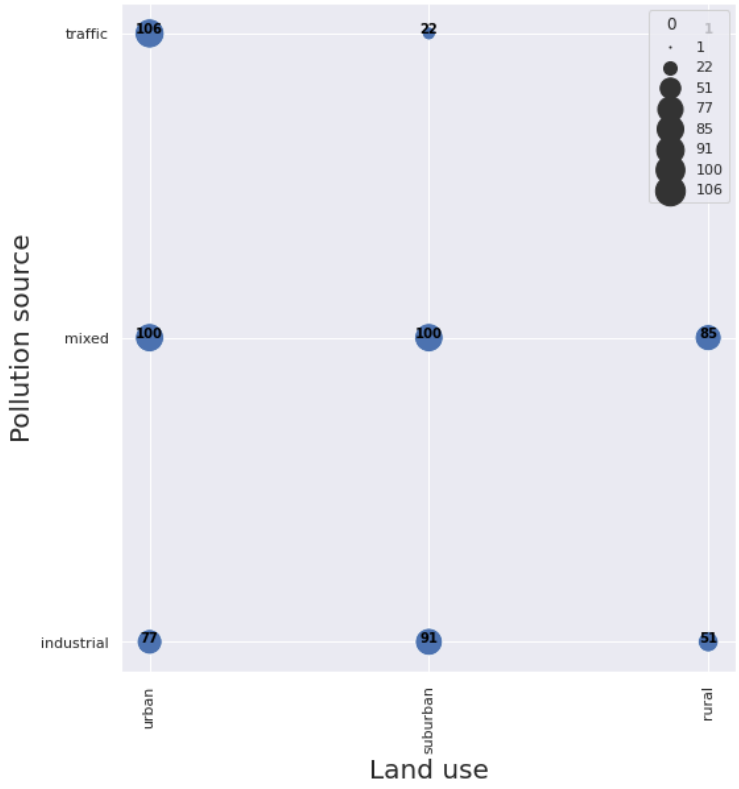


Fig. 4. Co-occurrence of the two stations' labels.

According to the official documents [20], these mobility data was collected through Call Detail Records (CDRs) from 13 million users of an unspecified mobile-phone carrier. Once anonymised, this dataset was used to infer representative mobility statistics at the nation level of the population of Spain and made publicly available as open data.

It is worth mentioning that this dataset captured the movement of people regardless of the means of transport used for their displacements. For that reason, we removed records whose distance *dist* was below or equal to 7 km. This is the average distance travelled by Europeans on foot or cycling for regular trips [1]. By mean of this filtering, we just kept records reporting trips that were more likely to be made by motorcycles, cars, trains or airplanes. These means of transport might have an impact on the air quality of a region as, in most cases, they emit pollutant particles.

Consequently, an MA $m \in \mathcal{M}$, where $\mathcal{M}$ is the set of all MAs, emits a univariate timeseries at hour h, $\mathcal{T}_h^m = \langle d_{h-n}^m \rightarrow d_{h-n+1}^m \rightarrow ... \rightarrow d_h^m \rangle$ where h_i^m is the sum of the incoming and outgoing human displacements of m at the i-th hour.

3.1.3 Weather conditions dataset. To collect the meteorological data, we made use of the *Reliable Prognosis* web service[4]. This platform provides an open repository with the meteorological conditions collected by multiple weather stations deployed at national and international airports worldwide. We extracted the temperature, wind speed and wind direction from weather stations on an hourly basis in all the Spanish airports on the platform. Fig. 5 shows the location of the

[4]https://rp5.ru/Weather_in_the_world

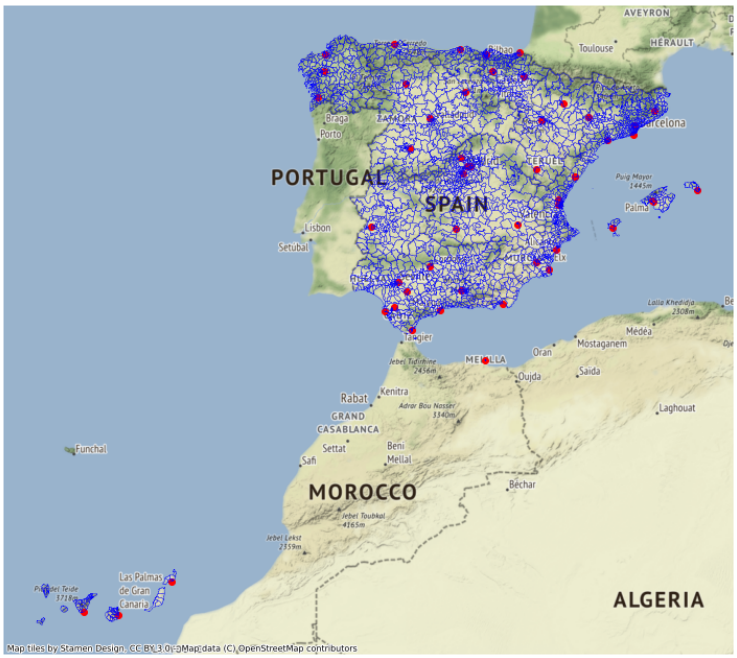


Fig. 5. Spatial boundaries of the Mobility Areas (MAs) modeling the human mobility as blue lines. The red points indicate the location of the airports used to collect nationwide weather conditions.

airports as red points. Let us call the set of all airports' weather stations as $\mathcal{W}$. Hence, a weather station $w \in \mathcal{W}$ generates a multivariate timeseries at time instant h, $\mathcal{T}_h^w = \langle\langle t_{h-n}^w, ws_{h-n}^w, wd_{h-n}^w\rangle \rightarrow \langle t_{h-n+1}^w, ws_{h-n+1}^w, wd_{h-n+1}^w\rangle \rightarrow .. \rightarrow \langle t_h^w, ws_h^w, wd_h^w\rangle\rangle$ where a tuple $\langle t_i^w, ws_i^w, wd_i^w\rangle$ represents the temperature (t), wind speed (ws) and wind direction (wd) measured by w at the i-th hour.

Finally, it is worth noticing that the 3 datasets considered in this work are public feeds that can be easily accessed by the research community.

3.2 Definition of the Heterogeneous Graph

Given the previous datasets, the prediction setting can be defined by means of three entities: the MAs, the air monitoring stations and the airports. Fig. 6 shows the heterogeneous graph that models the relationships among them. To begin with, the graph includes several *close-to* edges among the three entities so as to model their spatial relationships. In order to consider Tobler's first law of geography (*Everything is related to everything else, but near things are more related than distant things*) [19], each edge is weighted by the geographical distance among each pair of involved entities. For example, the link *close-to* between an airport w and MA m is labelled with the distance in kilometers between w and the spatial centroid of m. By means of this network of close-to relations, we allow the predictor to encode the air pollution patterns that not only occur in close locations but also in distant ones that might also arise in certain situations [12].

Besides, the *contain* edges allow encoding the relationship between the traffic activity of an MA and the air monitoring stations that are included within the geographical boundaries of the MA. Lastly, the *same land use* edges connect stations labelled with the same land use and the *same source* relationship connects stations whose primary pollution

source is also the same. With these two relationships, we allow the predictor to consider the similarities that might arise between monitoring stations that are distant in spatial terms but are enclosed in similar regions from a functional point of view.

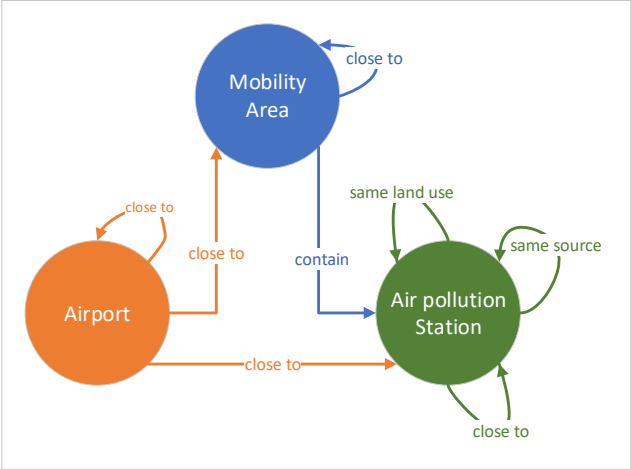


Fig. 6. Heterogeneous graph modeling the relationships among the MAs, the airports comprising the weather stations and the air monitoring stations.

3.3 Problem Definition

At this point, we can define the air-pollution forecasting task at hand as the following regression problem,

Given the hour h, and the set of timeseries $\mathcal{T}_h^{\mathcal{S}}, \mathcal{T}_h^{\mathcal{M}}, \mathcal{T}_h^{\mathcal{W}}$ comprising the timeseries of all the air monitoring stations $\mathcal{S}$, MAs $\mathcal{M}$ and weather stations $\mathcal{W}$, respectively, find a mapping function $\mathcal{F}$

$$\mathcal{F}(\mathcal{T}_h^{\mathcal{S}}, \mathcal{T}_h^{\mathcal{M}}, \mathcal{T}_h^{\mathcal{W}}) \rightarrow \mathcal{T}_{h+\Delta}^{\mathcal{S}}$$

where $\mathcal{T}_{h+\Delta}^{\mathcal{S}}$ are the predicted pollution levels of the particles measured by each station $m \in \mathcal{M}$ at the hour $h + \Delta$ where Δ (≥ 1) is the time horizon of the prediction.

3.4 Graph Neural Network Design

Given the heterogeneous graph and the regression problem defined in the previous two sections, we built the Graph Neural Network architecture shown in Fig. 7.

As observed, the proposed architecture comprises 4 different layers. In order to deal with the heterogeneous nature of the input graph, each layer includes a convolutional operator so that when multiple relations (edge types) point to the same destination node, the embedding of such a node is computed as the sum of its embedding for each edge type. This is depicted as the *HeteroConv* boxes in Fig. 7. By mean of this approach the nodes' embeddings are enriched at each step of the processing pipeline of the network with the information coming from all their connections.

More in detail, the first layer is in charge of capturing the temporal evolution of the timeseries generated by the MAs, air-pollution stations and airports' sensors. To do so, we apply an Integrated Graph Convolutional Long Short Term Memory (Conv-LSTM) model [21]. Basically, this model stacks a graph CNN and an LSTM cell. More in detail, the

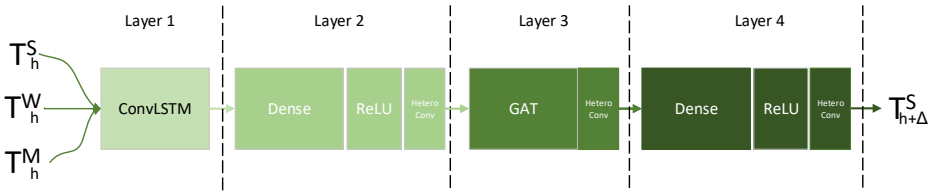


Fig. 7. Layer structure of the proposed GNN.

cell comprises three different gates, the update $z(t)$, the reset $r(t)$ and the memory-content $g(t)$. The computation of each gate is as follows:

$$
\begin{aligned}
x_i^{CNN} &= CNN_{\mathcal{G}}(\mathcal{G}_i^t), \\
i &= \sigma(W_{xi}\, x_i^{CNN} + W_{hi}\varepsilon_{i-1}^t + w_{ci} \odot c_{t-1} + b_i), \\
f &= \sigma(W_{xf}\, x_i^{CNN} + W_{hf}\varepsilon_{i-1}^t + w_{cf} \odot c_{t-1} + b_f), \\
c_t &= f_t \odot c_{t-1} + i_t \odot tanh(W_{xc} x_t^{CNN} + W_{hc}\varepsilon_{i-1}^t + b_c), \\
o &= \sigma(W_{xo}\, x_i^{CNN} + W_{ho}\varepsilon_{i-1}^t + w_{co} \odot c_t + b_o), \\
\varepsilon_i^t &= o \odot tanh(c_t)
\end{aligned}
\tag{1}
$$

where W_x and W_h are the weights of the fully connected layers, $b_{\{i,f,c,o\}}$ are the bias terms of each layer and $CNN_{\mathcal{G}}$ is the Chebyshev spectral graph convolutional operator [5]. To do so, the layer uses the edges of the heterogeneous graph defined in sec. 3.2. As a result, this model generates the embeddings with the latent representation of MAs, weather and air-pollution stations based on their temporal evolution. Next, these embeddings are passed through a first dense layer with the ReLU activation function. By means of this first dense layer we allow the model to learn relationships among pollutants and stations that are not directly encoded in the heterogeneous graph.

In the third layer, the resulting embeddings are processed by a Graph ATtention (GAT) layer [23]. This layer incorporates a multi-head attention mechanism that allows weighting the neighbours of a node based on their importance. Hence, the latent representation of a node v, ϵ_v giving K attention heads, is computed as

$$
\varepsilon_v = \sigma\Big(\frac{1}{K}\sum_{k=1}^{K}\sum_{u\in\mathcal{N}_v}\alpha_{vu}^k W^k \varepsilon_u\Big)
$$

where α_{vu}^k is the normalize coefficient and W^k is the linear-transformation weight matrix of the k-th attention mechanism. By means of this attention mechanism, the model is able to learn the importance of the links across the different types of edges. For example, it enables giving more importance to the *same-source* links between air-pollution stations with the same source of pollution exhibiting strong similarities in their pollution patterns despite the fact that they might be very distant in spatial terms. In addition to that, it allows giving higher importance to the *close-to* links between MAs and air-pollution entities in scenarios where there is a very strong relationship between the traffic activity and the air quality of a region.

Finally, the embedding generated by the GAT operator is processed by the fourth layer, which is a dense network and a ReLu activation function to eventually generate the prediction set $\mathcal{T}_{h+\Delta}^{\mathcal{S}}$.

3.5 Methodology for GNN generation

As discussed in sec. 3.1.1, one of the crucial aspects of this proposal is its handling of a heterogeneous setting, where not all stations are capable of detecting the entire set of pollutants in $\mathcal{P}$. To ensure accurate air pollution forecasting, we devised a methodology for determining the optimal combination of pollutants for each station. This involved a grid search approach that consisted of the following iterative steps:

(1) At the i-th iteration, we select a subset of pollutants $\mathcal{P}'_i \subset \mathcal{P}$ comprising up to 5 pollutants ($|\mathcal{P}'_i| \leq 5$).
(2) Next, we keep the air-pollution stations that are able to measure all the particles in $\mathcal{P}'_i$, giving raise to the subset $\mathcal{S}'_i \subset \mathcal{S}$. Let us call the tuple $\mathcal{R}_i = \langle \mathcal{P}'_i, \mathcal{S}'_i \rangle$ the i-th *partition* of the station infrastructure.
(3) Based on $\mathcal{R}_i$, we generate and train a GNN instance, following the architecture presented in sec. 3.4, for a particular time horizon Δ, $GNN^{\Delta}_{\mathcal{R}_i}$. In this case, the GNN considers all the MAs $\mathcal{M}$ and weather stations $\mathcal{W}$ of the setting.
(4) Finally, we evaluate the prediction error of $GNN^{\Delta}_{\mathcal{R}_i}$ for each particle $p \in \mathcal{P}'_i$, $Err_p^{\Delta,\mathcal{R}_i}$.
(5) We repeat steps 1-4 for each possible subset of $\mathcal{P}'$ given all the 5-combinations of $\mathcal{P}$, $\mathcal{P}' \in C(|\mathcal{P}|, 5)$.

The rationale of restricting the number of pollutants in the target subsets $\mathcal{P}'$ to 5 is twofold. Firstly, it avoids an explosion of experiments with the generation of a large number of combinations of pollutants. Furthermore, Fig. 3a shows that the most frequent number of pollutants measured by a station is 5. Therefore, this number provides a suitable trade-off between generating a rich palette of evaluation scenarios and a feasible computation cost.

On the basis of aforementioned search, let us call $\mathcal{R}^p$ to the set comprising all the station partitions $\mathcal{R}_i$ that include p among their target pollutants ($\mathcal{R}^p = \{\mathcal{R}_\alpha\} \mid p \in \mathcal{R}_\alpha.\mathcal{P}'_\alpha$). Next, given a pollutant p, its associated set $\mathcal{R}^p$ and a particular time horizon Δ, we extract the smallest subset of partitions $\mathcal{R}^{p,\Delta}_{min} = \{\mathcal{R}_\beta\} \subseteq \mathcal{R}^p$ that accomplishes 2 conditions: 1) the average error of their GNNs to predict p given Δ ($Err_p^{\Delta,\mathcal{R}_\beta}$) is as low as possible and 2) every air-pollution station able to measure p is included in at least one partition of $\mathcal{R}^p_{min}$ ($\bigcup \mathcal{R}_\beta.\mathcal{S}'_\beta = \mathcal{S}^p$ where $\mathcal{S}^p$ is the set of stations able to measure p). Finally, the GNNs associated to $\mathcal{R}^{p,\Delta}_{min}$, $\mathcal{G}^{\Delta}_p$ compose the set of model infrastructure able to infer the pollutant p with a Δ time horizon for all its stations in the whole infrastructure.

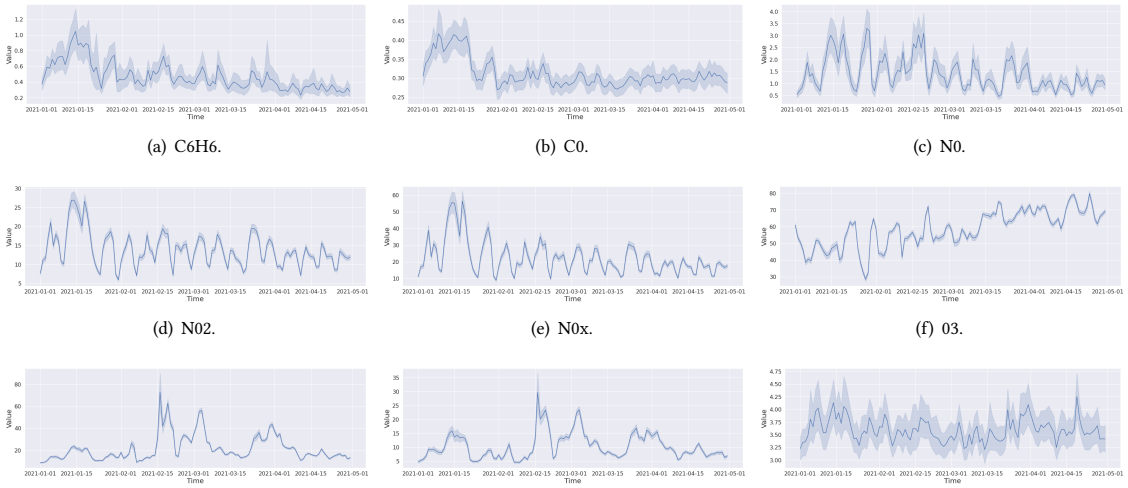
For the sake of clarity, we define a toy example with 4 different pollutants $\mathcal{P} = \langle p_1, p_2, p_3, p_4 \rangle$, where the pollutant p_2 is measured by 5 different stations, $\mathcal{S}^{p_2} = \langle s_1, s_2, s_3, s_4, s_5 \rangle$ and a time horizon Δ=12h. Given this setting, let us assume that $\mathcal{R}^{p_2}$ comprises 3 different partitions $R_{\alpha 1} = \langle \langle p_2 \rangle, \langle s_2, s_3, s_4, s_5 \rangle \rangle$, $R_{\alpha 2} \langle \langle p_1, p_2, p_3 \rangle, \langle s_2, s_5 \rangle \rangle$ and $R_{\alpha 3} \langle \langle p_2, p_4 \rangle, \langle s_1, s_3, s_4 \rangle \rangle$ with the following accuracy of their GNNs, $Err_{p_2}^{12,\mathcal{R}_{\alpha 1}} = 0.30$, $Err_{p_2}^{12,\mathcal{R}_{\alpha 2}} = 0.15$ and $Err_{p_2}^{12,\mathcal{R}_{\alpha 3}} = 0.25$. Therefore, $\mathcal{R}^{p,\Delta}_{min}$ will comprise partitions $R_{\alpha 2}$ and $R_{\alpha 3}$ as they have the lowest error and cover all the stations in $\mathcal{S}^{p_2}$. As a result, $\mathcal{G}^{12}_{p_2} = \langle GNN^{12}_{\mathcal{R}_{\alpha 2}}, GNN^{12}_{\mathcal{R}_{\alpha 3}} \rangle$. These would be the two GNNs required to predict p_2 in all stations of the infrastructure.

4 EVALUATION OF THE PROPOSAL

In order to evaluate our proposal, we collected a data corpus from the three datasets described in sec. 3.1 covering a 4-month period from January 1st, 2021 to April 30th, 2021. For this period, Fig. 8 shows the global timeseries of the 9 target pollutants, Fig. 9 the average number of trips per MA and hour and Fig. 10 the timeseries of the weather features.

4.1 Evaluation Metrics

Regarding the metrics to evaluate our approach, the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) [25] are two of the most common metrics used to measure accuracy for continuous variables. They are suitable



(a) C6H6. (b) C0. (c) N0.

(d) N02. (e) N0x. (f) 03.

(g) PM10. (h) PM25. (i) SO_2.

Fig. 8. Timeseries of each pollutant considering all the stations $\mathcal{S}$ during the evaluation period. The dark blue line indicates the average value of all the stations whereas the bluish area around the line indicates the 95% confidence interval.

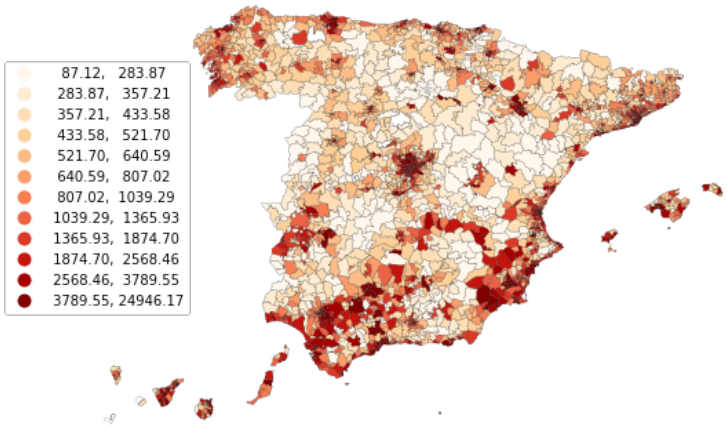
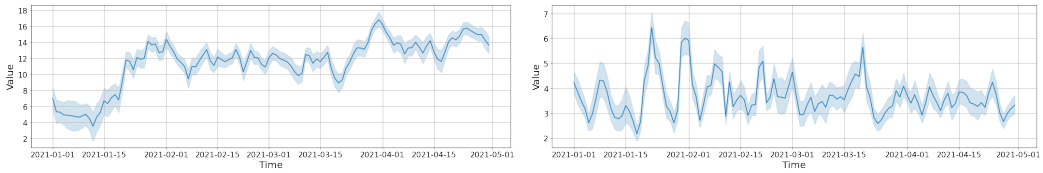


Fig. 9. Average number of hourly incoming and outgoing trips per MA during the evaluation period.

for model comparisons as they express average model prediction error in the units of the variable of interest. Their definition is as follows:

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

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



(a) Temperature (in C^0). (b) Wind speed (in meters per second).

Fig. 10. Timeseries of the ambient temperature and wind speed considering all the airports' weather stations $\mathcal{W}$ during the evaluation period. The dark blue line indicates the average value of all the stations whereas the bluish area around the line indicates the 95% confidence interval.

where, for our experiment, y_i is the real pollution value of a particular pollutant at a certain hour, $\hat{y}_i$ is the predicted level at this same scale and n is the number of observations. These two measures are valid to represent the error $Err_p^{\Delta,\mathcal{R}_i}$ in the coverage algorithm described in sec. 3.5.

4.2 Evaluation Settings

Concerning the hyperparameters used in the evaluation of the proposal, Table 2 comprises the most relevant ones related to the training strategy and each of the 4 operators of the GNN according to sec. 3.4.

Table 2. Model parameters for the experiments

Type	Parameter	Value
Training	Training rate	0.95
	Loss	Mean Squared Error (MSE)
	Learning factor	0,01
	Weight decay	0.0005
	Optimizer	Adam
	Early Stop	patience: 5, min_delta: 0.001
	Num. of epochs	40
Conv-LSTM	Output size	128
	Num. cells	2
Dense - layer 2	Input size	128
	Output size	128
GAT	Num. of heads	4
	Output size	128*4=512
Dense - layer 4	Input size	512
	Output size	Num. of target particles

4.3 Baseline

As baseline to compare our approach, we have made use of a model that combines a Convolutional Neural Network [16] and a Long Short-Term Memory [22] forecasting models (CNN-LSTM). It consists of a first data input layer, followed by the usual CNN architecture (i.e. a Convolutional layer, a Max Pooling layer and a final Flatten layer). The last Flatten

layer is connected to an LSTM layer to capture possible trends over time. Finally, the model has a last dense layer in charge of generating all the predictions.

For the generation of the pollutant forecasts, a total of 633 instances of the CNN-LSTM model have been created, one for each station object of this study. Thus, this baseline provides a very fine-grained ensemble of models to be compared with our approach. In that sense, we propose a more *coarse-grained* ensemble where a GNN covers an aggregation of stations instead of on a single one like the baseline.

4.4 Results Discussion

Table 3 shows the average MAE and RMSE per particle and Δ of the $\mathcal{G}_p^\Delta$ sets and the alternative CNN-LSTM models, respectively. Note that the range of time horizons (12-192h) in our system is larger than the majority of the configurations usually proposed in the literature, involving 4 [7], 24 [10] or 48 [13] hour predictions.

The results show that our approach clearly outperformed the baseline for almost all the particles and time horizons. For example, the average MAE for the PM_{10} particle for a 96-hour prediction was 11.347 for our approach and 39.726 for the CNN-LSTM alternative. For this specific result, it is worth mentioning that we compared the results of 302 of CNN-LST models (the number of stations able to measure PM_{10} particles according to Fig. 3b) and 5 GNNs (the number of models included in the $\mathcal{G}_{PM10}^{96}$ set). Besides, the RMSE of our approach to predict NO_2 particles for a 192-hour horizon was 4.590, whereas the RMSE for the CNN-LSTM alternative was 9.399. In this case, the cardinality of $\mathcal{G}_{NO_2}^{192}$ was 8 and the number of CNN-LSTM models were 466.

The results of both tables show that we obtained more accurate results with a significant smaller number of models. This has important implications in operational terms, as the deployment and management of a solution involving less models is easier than a solution needing more instances.

Table 3. Average prediction error and standard deviation of the sets $\mathcal{G}_p^\Delta$ and the CNN-LSTM model for each combination of pollutant p and time horizon Δ.

MODEL		MAE					RMSE				
	T	12h	24h	48h	96h	192h	12h	24h	48h	96h	192h
GNN	NO	0.421±0.017	0.352±0.006	0.399±0.008	0.458±0.009	0.497±0.013	0.505±0.017	0.409±0.011	0.455±0.013	0.509±0.005	0.567±0.012
	C6H6	0.096±0.02	0.089±0.022	0.078±0.014	0.086±0.03	0.101±0.029	0.126±0.032	0.113±0.034	0.099±0.018	0.11±0.043	0.128±0.045
	CO	0.037±0.002	0.034±0.003	0.042±0.003	0.059±0.001	0.081±0.0	0.042±0.002	0.04±0.005	0.047±0.005	0.062±0.001	0.089±0.0
	NO2	4.278±0.293	3.511±0.011	3.758±0.083	3.812±0.056	3.62±0.316	5.092±0.526	4.476±0.037	4.557±0.212	4.578±0.265	4.59±0.505
	NOx	6.373±0.438	4.528±0.0	5.134±0.208	5.474±0.203	5.317±0.146	7.677±1.007	5.943±0.0	6.463±0.25	6.831±0.279	6.688±0.287
	O3	20.53±0.43	13.59±0.519	11.398±0.579	14.751±0.081	13.239±1.328	22.759±0.36	15.607±0.64	13.018±0.849	16.886±0.096	15.548±1.491
	PM10	5.322±0.0	4.705±0.0	7.281±0.229	11.347±0.185	11.427±0.336	6.028±0.0	5.322±0.0	9.012±0.154	12.603±0.196	12.837±0.415
	PM25	3.647±0.379	2.78±0.398	3.74±0.925	6.367±2.512	5.964±1.772	4.046±0.492	3.258±0.598	4.32±1.324	7.892±3.899	7.288±2.812
	SO2	0.809±0.029	0.872±0.0	0.81±0.032	0.771±0.207	1.223±0.05	1.154±0.04	1.185±0.0	0.996±0.029	0.938±0.223	1.578±0.069
CNNLSTM	NO	2.173±1.836	2.109±1.687	2.177±1.52	2.065±1.168	2.109±1.041	2.34±1.854	2.284±1.69	2.503±1.733	2.348±1.346	2.506±1.164
	C6H6	0.728±0.708	0.724±0.686	0.728±0.676	0.737±0.687	0.71±0.634	0.785±0.711	0.8±0.704	0.843±0.777	0.895±0.93	0.852±0.776
	CO	0.178±0.157	0.176±0.151	0.181±0.158	0.219±0.454	0.298±1.149	0.193±0.158	0.192±0.151	0.202±0.171	0.266±0.736	0.352±1.427
	NO2	6.651±4.365	6.131±3.818	6.185±3.887	6.169±3.737	7.206±7.42	8.019±5.174	7.502±4.47	7.67±4.69	7.676±4.495	9.399±10.099
	NOx	17.134±12.671	15.769±11.722	15.595±11.767	15.785±15.19	18.113±24.078	20.337±15.719	18.742±13.584	18.643±13.442	18.985±19.09	23.377±38.773
	O3	20.252±10.391	16.474±6.825	15.661±6.897	17.743±24.646	21.628±56.336	22.578±10.89	19.621±7.697	18.933±8.316	21.796±38.766	26.35±69.343
	PM10	11.573±9.03	12.763±15.084	18.675±85.206	39.726±264.455	68.627±449.861	12.964±9.618	14.679±20.607	22.891±117.517	52.003±364.298	85.839±544.705
	PM25	7.537±6.019	7.808±6.44	10.037±25.876	13.436±53.22	22.161±89.406	8.176±6.017	8.725±7.368	11.879±35.914	15.799±63.247	28.644±118.508
	SO2	4.844±5.188	4.843±5.074	4.999±6.54	5.871±15.059	7.227±25.446	5.181±5.301	5.324±5.423	5.647±8.376	7.048±21.843	8.934±33.159

Furthermore, Fig. 11 shows the cardinality of the sets $\mathcal{G}_p^\Delta$ for each pollutant-Δ tuple. For example, the 48-h prediction of the NO_2 pollutant required 3 different GNNs. In this case, we can observe 2 different patterns with respect to the number of models.

On the one hand, some pollutants required a largest number of GNNs to cover all its stations as long as the time horizon increased. This is the case of NO2, PM_{10} and $PM_{2.5}$. This positive correlation was also observed in global terms, as shown in the *global* column of Fig. 11. Hence, when we increase the number of GNNs for a pollutant, it causes that

each GNN focuses on a smaller number of stations. This pattern indicates that some pollutants require more specific GNNs when the time horizon increases. Therefore, the relationships established among MAs, stations and airports are constrained to a more limited number of elements. This allows us to remove certain connections from the model that are not meaningful for the prediction horizon at hand.

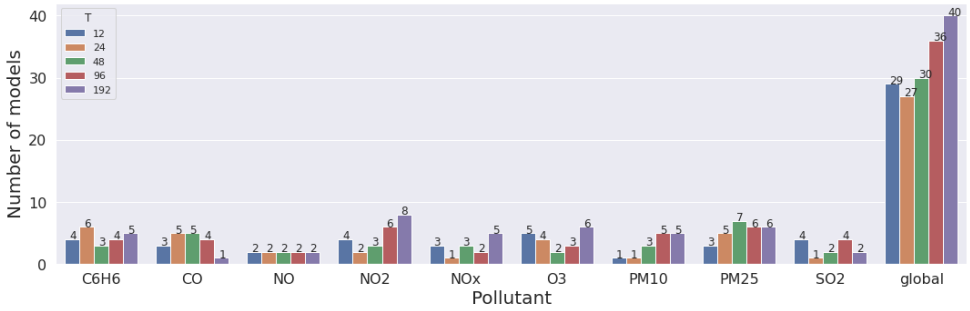


Fig. 11. Number of required GNNs to predict a particular pollutant for a time horizon configuration. Each bar is labelled with its actual value. The *global* column indicates the total number of GNNs to cover all the stations for a particular time horizon.

On the other hand, other pollutant exhibited a more stable number of GNNs regardless the prediction horizon such as NO or C6H6. This reflects that the prediction of these particles does not require more specific GNNs from the point of view of the target stations. As a result, they are more *scalable* inputs in order to deploy GNNs covering a wide range or time horizon configurations.

Fig. 12 shows the average number of inputs of each $\mathcal{G}_p^{\Delta}$ model. It is worth noticing the fact that most of the models do not require a large number of particles to predict one of them. In fact, they just need bi-variate timeseries to perform the prediction. For instance, the GNNs to predict the SO_2 particle for 48 and 192 hours actually used the timeseries from this particle and another one to perform the prediction. These results suggest that a *divide-and-conquer* approach with multiple models targeting a limited number of pollutants is more reliable than a solution comprising few and multi-variate models.

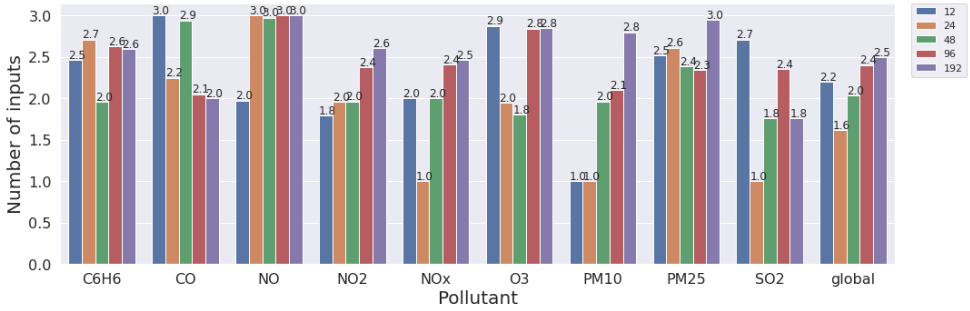


Fig. 12. Average number of inputs of the GNNs to predict a particular pollutant for a time horizon configuration. Each bar is labelled with its actual value. The *global* column indicates the average number of inputs of the models regardless of the particle for a particular time horizon.

Elaborating on this finding, Table 13 shows the co-occurrence between pairs of pollutants. Firstly, it is observed that the most frequent co-occurrences are between pollutants whose estimated primary sources are different. This is the case of the prediction of $PM_{2.5}$ –a pollutant mainly caused by road-traffic emissions– whose most frequent co-occurrence is SO_2 –a pollutant mainly caused by industry emissions. A similar pattern is also observed between O_3 and NO_2. On the contrary, other pollutants such as NO_2, $PM_{2.5}$ or SO_2 are more frequently predicted using other pollutants with similar sources (CO, NO_2 and O_3, respectively). This disparity of frequency of the co-occurrences between pollutants and their estimated sources reinforces the idea of developing forecasting solutions defined as an ensemble of predictors, each one considering a particular set of specific relationships between pollutants.

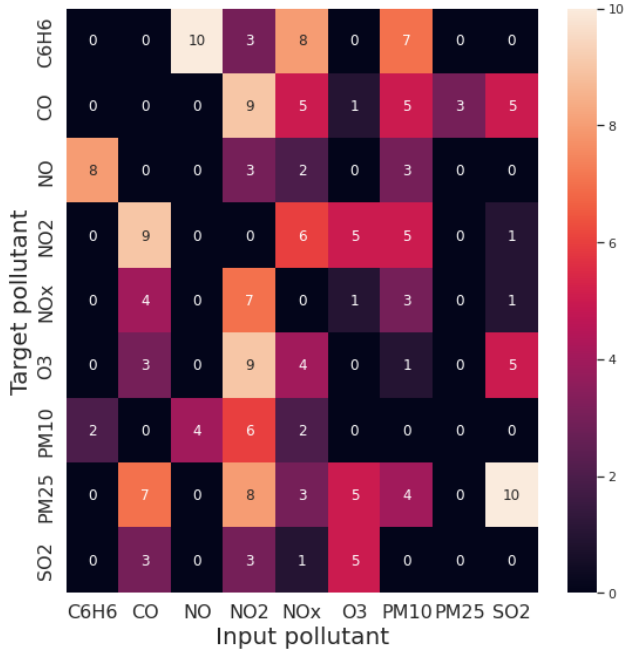
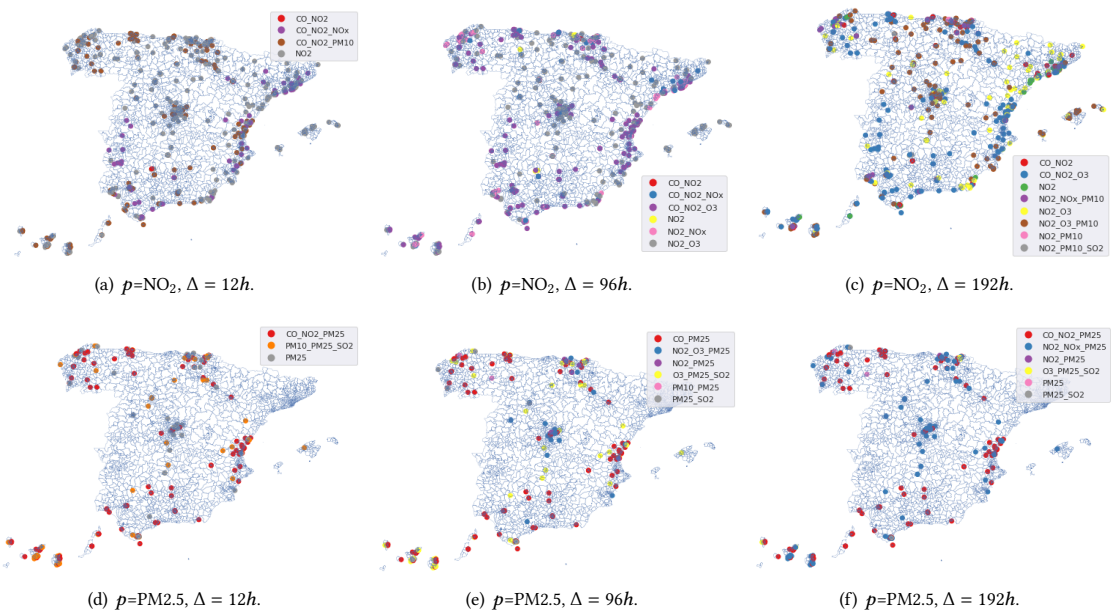


Fig. 13. Co-ocurrence of pollutants as input of the GNNs. The rows indicate the target pollutant to be predicted by the model and the columns indicate another pollutant that is used by the GNN as input to improve the prediction accuracy.

Finally, Fig. 14 shows some interesting patterns of the spatial distribution of the GNNs for certain tuples of particles and time horizons. For example, Fig. 14a shows that the GNN taking as input the particles CO_2, NO_2 and O_3 (purple points) mainly processes the stations located at the exterior boundary of Spain. Moreover, the red points in Figs. 14d-f show a persistent GNN covering the stations located in the East coast and South and North-West regions of Spain in order to predict the $PM_{2.5}$ level. This GNN always uses CO as a secondary input for its forecasting task, revealing that certain stations should be processed together despite being quite far allocated in spatial terms. This is due to the fact that the GNN architecture (sec. 3.4) is able to capture other latent relationships among stations that go beyond their spatial distance such as their surrounding land use or their most likely pollution source.



(a) p=NO_2, $\Delta = 12h$. (b) p=NO_2, $\Delta = 96h$. (c) p=NO_2, $\Delta = 192h$.

(d) p=PM2.5, $\Delta = 12h$. (e) p=PM2.5, $\Delta = 96h$. (f) p=PM2.5, $\Delta = 192h$.

Fig. 14. Spatial distribution of the $\mathcal{G}_{\Delta}^{p}$ set for different combinations of particles and time horizons.

5 CONCLUSIONS AND FUTURE WORK

Air pollution forecasting plays a crucial role in addressing current environmental and public health challenges. By providing advanced knowledge of air pollutant levels, it could be possible to take proactive measures to mitigate the adverse effects of pollution. Graph Neural Networks (GNNs) have emerged as one of the most advanced techniques for building air pollution prediction models. However, existing solutions often focus solely on spatial distance relationships among stations, disregarding other essential spatial and feature-based contextual factors. Additionally, they assume a homogeneous setting where all stations have the same pollutant measurement capabilities.

In this work, we introduce a GNN framework capable of capturing relevant air monitoring station's features such as the land use of their locations and their primary pollution sources. Moreover, the GNN also receives data on human mobility regarding to traffic pollution as well as weather data including temperature and wind speed and direction. Thus, this framework enables a more comprehensive representation of the complex dynamics and relationships in air pollution data. Furthermore, we developed a methodology specifically designed to address heterogeneous settings within the GNN architecture, accommodating stations with different measurement capabilities. As a result, our solution can measure up to 9 pollutants, including not only the most frequent ones in the literature, such as PM_{10} or O_3, but also less studied pollutants like benzene (C6H6).

The evaluation of our proposal was conducted on a nationwide scale in Spain, utilizing 4 months of hourly data collected from 633 heterogeneous air monitoring stations. The results demonstrated that the proposed model was able to predict pollutants such as PM_{10} and NO_2 over time horizons of 96 and 192 hours, surpassing the performance of a baseline model. This is an outstanding result taking into account that the reviewed literature typically focused on shorter time horizons of 24/48 hours. Moreover, some interesting spatial distributions emerged from the resulting GNNs,

which corresponded to the land use of the air monitoring stations, as well as various relationships of co-occurrence of pollutants linked to their primary sources. The findings of this work may aid urban planning and policy-making by informing strategies for emission reduction, traffic management, and the implementation of specific interventions in any area of a country.

As future work, we are currently exploring the feasibility of implementing transfer learning techniques within our proposed framework. This involves leveraging the knowledge acquired from specific patterns of spatial distribution captured by the GNNs and applying it to regions or cities with limited air pollution data. Another line of research is the analysis of the deployment and scalability of our model in air monitoring stations by addressing the computational challenges found in these systems. In this manner, the operators of these system could develop new applications based on the predicted air pollution data.

ACKNOWLEDGMENTS

Financial support for this research has been provided under grant PID2020-112827GB-I00 funded by MCIN/AEI/10.13039/501100011033. It is also partially granted by the "EMERGIA" programme, funded by the Junta de Andalucía through the grant EMC21_004171.

REFERENCES

[1] 2022. *Passenger mobility statistics.* Technical Report. Eurostat.

[2] Patricia Arroyo, José Luis Herrero, José Ignacio Suárez, and Jesús Lozano. 2019. Wireless sensor network combined with cloud computing for air quality monitoring. *Sensors* 19, 3 (2019), 691.

[3] Zena A Aziz Aziz and Siddeeq Y Ameen Ameen. 2021. Air pollution monitoring using wireless sensor networks. *Journal of Information Technology and Informatics* 1, 1 (2021), 20–25.

[4] Kevin Cromar and Noussair Lazrak. 2023. Risk communication of ambient air pollution in the WHO European Region: review of air quality indexes and lessons learned. (2023).

[5] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems* 29 (2016).

[6] Zhenni Ding, Huayou Chen, Ligang Zhou, and Zicheng Wang. 2022. A forecasting system for deterministic and uncertain prediction of air pollution data. *Expert Systems with Applications* 208 (2022), 118123.

[7] Ao Dun, Yuning Yang, and Fei Lei. 2022. Dynamic graph convolution neural network based on spatial-temporal correlation for air quality prediction. *Ecological Informatics* 70 (2022), 101736.

[8] Shaohua Fan, Junxiong Zhu, Xiaotian Han, Chuan Shi, Linmei Hu, Biyu Ma, and Yongliang Li. 2019. Metapath-guided heterogeneous graph neural network for intent recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2478–2486.

[9] Aysenur Gilik, Arif Selcuk Ogrenci, and Atilla Ozmen. 2022. Air quality prediction using CNN+ LSTM-based hybrid deep learning architecture. *Environmental science and pollution research* (2022), 1–19.

[10] Jindong Han, Hao Liu, Haoyi Xiong, and Jing Yang. 2022. Semi-supervised air quality forecasting via self-supervised hierarchical graph neural network. *IEEE Transactions on Knowledge and Data Engineering* (2022).

[11] Azim Heydari, Meysam Majidi Nezhad, Davide Astiaso Garcia, Farshid Keynia, and Livio De Santoli. 2021. Air pollution forecasting application based on deep learning model and optimization algorithm. *Clean Technologies and Environmental Policy* (2021), 1–15.

[12] Rosaria Ignaccolo, Stefania Ghigo, and Stefano Bande. 2013. Functional zoning for air quality. *Environmental and ecological statistics* 20 (2013), 109–127.

[13] Ditsuhi Iskandaryan, Francisco Ramos, and Sergio Trilles. 2023. Graph Neural Network for Air Quality Prediction: A Case Study in Madrid. *IEEE Access* 11 (2023), 2729–2742.

[14] Xue-Bo Jin, Zhong-Yao Wang, Jian-Lei Kong, Yu-Ting Bai, Ting-Li Su, Hui-Jun Ma, and Prasun Chakrabarti. 2023. Deep spatio-temporal graph network with self-optimization for air quality prediction. *Entropy* 25, 2 (2023), 247.

[15] Gamze Kurnaz and Alparslan Serhat Demir. 2022. Prediction of SO2 and PM10 air pollutants using a deep learning-based recurrent neural network: Case of industrial city Sakarya. *Urban Climate* 41 (2022), 101051.

[16] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. 2021. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems* (2021).

[17] Nilesh N Maltare and Safvan Vahora. 2023. Air Quality Index prediction using machine learning for Ahmedabad city. *Digital Chemical Engineering* 7 (2023), 100093.

[18] Concettina Marino, Antonino Nucara, Maria Francesca Panzera, and Matilde Pietrafesa. 2022. Assessment of the road traffic air pollution in urban contexts: a statistical approach. *Sustainability* 14, 7 (2022), 4127.

[19] Harvey J Miller. 2004. Tobler's first law and spatial analysis. *Annals of the association of American geographers* 94, 2 (2004), 284–289.

[20] Movilidad y Agenda Urbana Secretaría de Estado de Transportes. 2020. *Análisis de la movilidad en España con tecnología Big Data durante el estado de alarma para la gestión de la crisis del COVID-19.* Technical Report. Ministerio de Transportes, Movilidad y Agenda Urbana.

[21] Youngjoo Seo, Michaël Defferrard, Pierre Vandergheynst, and Xavier Bresson. 2018. Structured sequence modeling with graph convolutional recurrent networks. In *Neural Information Processing: 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, December 13-16, 2018, Proceedings, Part I 25*. Springer, 362–373.

[22] Greg Van Houdt, Carlos Mosquera, and Gonzalo Nápoles. 2020. A review on the long short-term memory model. *Artificial Intelligence Review* 53 (2020), 5929–5955.

[23] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio, et al. 2017. Graph attention networks. *stat* 1050, 20 (2017), 10–48550.

[24] Xiao Wang, Deyu Bo, Chuan Shi, Shaohua Fan, Yanfang Ye, and S Yu Philip. 2022. A survey on heterogeneous graph embedding: methods, techniques, applications and sources. *IEEE Transactions on Big Data* 9, 2 (2022), 415–436.

[25] C. J. Willmott and K. Matsuura. 2005. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research* 30, 1 (2005), 79–82. cited By (since 1996)149.

[26] Carlos Zafra, Yenifer Ángel, and Eliana Torres. 2017. ARIMA analysis of the effect of land surface coverage on PM10 concentrations in a high-altitude megacity. *Atmospheric Pollution Research* 8, 4 (2017), 660–668.

[27] Nur'atiah Zaini, Lee Woen Ean, Ali Najah Ahmed, and Marlinda Abdul Malek. 2022. A systematic literature review of deep learning neural network for time series air quality forecasting. *Environmental Science and Pollution Research* (2022), 1–33.

[28] Shuang Zhao, Shiliang Liu, Xiaoyun Hou, Yongxiu Sun, and Robert Beazley. 2021. Air pollution and cause-specific mortality: a comparative study of urban and rural areas in China. *Chemosphere* 262 (2021), 127884.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009