

# Transfer learning with global satellite data to predict localised nitrogen dioxide levels

By  
**Finn Gueterbock**

supervised by

Dr R. Santos-Rodriguez and Dr J. Clark



Department of Engineering Mathematics  
School of Computer Science, Electrical and Electronic Engineering, and Engineering Maths  
University of Bristol

A project report submitted to the University of Bristol in accordance with the requirements of  
the degree of MASTERS OF ENGINEERING in the Faculty of Engineering.

August, 2023

# Abstract

This report proposes a novel approach for predicting NO<sub>2</sub> concentrations at unseen locations in Bristol using satellite air quality and meteorological data, with the ultimate goal of creating a ‘virtual sensor’ for areas with limited air quality monitoring resources. Our approach utilises the GraphSAGE framework for inductive learning on graphs, combining autoregression and transfer learning with data from London in order to improve performance on the limited data available in Bristol. The proposed method is shown to have a 17% reduction in Normalised Root Mean Squared Error (NRMSE) compared to current models when evaluated on the same dataset.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Literature review</b>	<b>4</b>
<b>3</b>	<b>Data</b>	<b>5</b>
3.1	Surface NO <sub>2</sub> Measurements . . . . .	5
3.2	Additional Features . . . . .	6
3.2.1	Satellite Data . . . . .	6
3.2.2	Meteorological Data . . . . .	7
3.2.3	Road Data . . . . .	7
3.3	Data Pre-processing . . . . .	7
3.4	Data Exploration . . . . .	7
3.4.1	Analysis of data suitability for transfer learning . . . . .	8
<b>4</b>	<b>Models</b>	<b>10</b>
4.1	Convolutional Neural Network Model . . . . .	12
4.1.1	CNN transfer learning . . . . .	13
4.2	Autoregressive GraphSAGE Model . . . . .	16
4.2.1	GraphSAGE . . . . .	16
4.2.2	Training process . . . . .	18
4.2.3	Model parameter selection . . . . .	19
<b>5</b>	<b>Results and Discussion</b>	<b>23</b>
<b>6</b>	<b>Conclusions and Future Directions</b>	<b>25</b>

# 1 Introduction

Air pollution is one of the leading causes of death worldwide, killing around 7 million people each year [1]. Furthermore, in Europe it was found that more than 75 000 premature deaths a year could have been prevented by reducing the level of nitrogen dioxide ( $\text{NO}_2$ ) to its lowest recorded levels [2]. As such, increasing the availability of air quality data is of great relevance to public health authorities. However, global air quality monitoring coverage is limited. In 2019, it was found that only 24 countries had more than 3 monitors per million inhabitants, while 141 had no particulate matter (PM) monitoring stations at all [3].

$\text{PM}_{2.5}$  (particulate matter less than  $2.5\mu\text{m}$  in diameter) and  $\text{NO}_2$  are the most dangerous air quality indicators to human health. In Bristol,  $\text{PM}_{2.5}$  and  $\text{NO}_2$  cause around 300 deaths a year, representing about 8.5% of total deaths in the city [4]. However, there are currently only 8 active  $\text{NO}_2$  monitoring sites and only 3  $\text{PM}_{2.5}$  sensors across Bristol, making it difficult to measure the effect of any policies to improve air quality in the area.

Meanwhile, access to satellite air quality data has become abundant in recent years, enabling near real-time global data collection and analysis. Satellite data has been used for monitoring of air quality [5] [6], however, the results from these studies lack spatial resolution and cannot currently replace localised ground-based measurements.

This project aims to increase the availability and spatial resolution of air quality data, utilising widely-available satellite data and meteorological features along with ground-based air quality measurements, working towards an accessible model which could be deployed to regions lacking infrastructure and resources for costly localised measurements. This will take the form of a model that, for some specified location, takes satellite and meteorological data and simulates the measurements that would be recorded by a ground-based  $\text{NO}_2$  monitor, i.e. a ‘virtual sensor’. Such a model would enable public health authorities to make more informed decisions, thereby improving air quality and reducing the number of premature deaths caused by air pollution.

These goals will be achieved by answering the following questions:

1. Can a model be built to accurately recreate the output of an  $\text{NO}_2$  monitoring station at an unseen location?
2. Can data from other cities be used to improve such a model?

While this project focuses on modelling  $\text{NO}_2$  concentrations in a developed city with a pre-existing network of air quality monitoring stations, the process of using data from a city with more resources, alongside the use of data from low cost air quality sampling schemes is hoped to provide insights into how this approach can be adapted and applied in regions with limited resources.

## 2 Literature review

In order to increase the spatial resolution of air quality data, some form of interpolation must be performed to fill in the gaps. A typical approach to geospatial interpolation is called kriging [7], a method developed in the 1960s which produces a distance-weighted Gaussian approximation of values between two points. Kriging is useful in that it gives a distribution of potential values, but is limited by its Gaussian assumptions and thus is best used for approximating terrain. Neural networks do not rely on Gaussian assumptions, and when working with spatial data, it is common to utilise the network-like structures between data points in the form of Graph Neural Networks (GNNs). In this way, a network can learn trends in the data based on the physical distance between each of the data points, enabling complex patterns to be learned [8].

Focusing on predicting spatiotemporal trends in PM2.5 concentrations, Muthukumar et al. [9] used a Graph Convolutional Network (GCN) to interpolate meteorological data to form a dense image-like graph of meteorological features. They then fed these feature representations, along with satellite and ground-based PM2.5 data as a time series into a Convolutional Long Short-Term Memory network (ConvLSTM) to learn and predict how these evolve over time. They found that they could accurately predict PM2.5 sensor readings for up to 10 days in the future, demonstrating that it is possible to predict particulate matter levels given a large enough area from which to learn.

In order to reliably predict ground-based air quality indicators, we need to recognise the multitude of factors that they can depend on, along with the relationships between them. Aerosol Optical Depth (AOD) is a unitless measure of aerosols (such as smoke or dust particles) that are present between a satellite and the ground. Surface level PM2.5 values have been shown to be ‘correlated significantly’ with AOD measured by geostationary satellites [10], but this relationship is highly dependent on seasonal changes in the weather. NO<sub>2</sub> and PM2.5 levels can also be affected by meteorological factors such as air temperature, relative humidity and wind speed. Yang et al. [11] showed that these relationships vary depending on the seasons. They found that across cities in China, air temperature was negatively correlated with PM2.5 in autumn, but positively correlated in winter, and that relative humidity had a strong positive correlation in winter and spring, which was weaker in summer and autumn.

Regarding the prediction of NO<sub>2</sub> from satellite and weather data, Masih et al.[12] applied the Random Forest algorithm to predict the atmospheric concentration of NO<sub>2</sub> and found that its performance was better than other two data mining approaches. Several studies have also explored the use of neural network models for NO<sub>2</sub> prediction. For instance, Li et al.[13] used a geographically and temporally weighted regression in a nonlinear neural network framework to estimate regional NO<sub>2</sub> via space-time neural networks, finding that this model outperformed other models. Gardner et al. [14] used multilayer perceptron neural networks to predict hourly NO<sub>x</sub> and NO<sub>2</sub> concentrations in urban air in London for individual locations. They found that these models were capable of resolving complex patterns of source emissions without explicit external guidance, but made no attempt at building a generalised model for unseen locations.

Ghahremanloo et al.[15] used deep learning models to estimate daily ground-level NO<sub>2</sub> concentrations from remote sensing data. They found that the CNN model was superior to other well-known machine learning and regression models in the field. Finally, Liu et al.[16] used a LSTM model for daily NO<sub>2</sub> concentration forecasting and found that the maximum temperature and zone improved the prediction accuracy the most.

Since the ground-based air quality data in Bristol is limited, we must consider methods to increase the generalisability of a model trained on this data, enabling it to predict air quality in places it has not seen before. One common method of doing this is data augmentation [17], which can help a model avoid overfitting by expanding a dataset with new points generated from the existing dataset. This method has been used to increase the variance in satellite based air quality data [18] in order to avoid overfitting, however it does not ensure that a model will be able to generalise to other locations as the data is still coming from the same source. It is helpful to notice here that Bristol is not the only city in the UK with ground-based air quality monitors; London, for example, has a large network of NO<sub>2</sub> and PM2.5 monitors. Through a process called transfer learning, a model can be trained on data from London, and then fine-tuned on data from Bristol in an effort to increase the generalisability and performance of the model. Due to its flexibility and the availability of data from other cities, this is the approach that will be taken in this project.

The use of transfer learning has been increasingly explored for air quality prediction recently. For instance, Yadav et al.[19] used deep transfer learning on satellite imagery to improve air quality estimates in developing nations. They showed that their CNN model trained on a city with lots of data could be fine-tuned to perform well in a developing nation, predicting NO<sub>2</sub> at a resolution of 200m. Ma et al.[20] improved air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques. They proposed a bi-directional LSTM model, which had smaller errors when transfer learning was used, especially for larger temporal resolutions.

While it is clear that using satellite and meteorological data to predict ground based NO<sub>2</sub> concentrations is not a completely new idea, previous work has focused on predicting NO<sub>2</sub> over vast areas of land with relatively low spatial resolution. However, the availability of high spatial resolution data is crucial as it can offer valuable insights into the sources and patterns of NO<sub>2</sub> pollution, enabling informed targeted mitigation measures. This, therefore, is the gap that we aim to fill in this study.

## 3 Data

### 3.1 Surface NO<sub>2</sub> Measurements

For the ground-based air quality measurements in the Bristol area, the Air Quality Data Continuous dataset was used through the Open Data Bristol API [21], which contains hourly air

quality information for 19 different locations across Bristol, from 1993 - present, as shown in Figure 1. For London, we use data from the London Air Quality Network [22], which provides hourly NO<sub>2</sub> readings from 112 locations across Greater London. Over the period of time for which satellite data is available, there are a total of 246 572 data points in Bristol, compared to 4 182 699 in London.

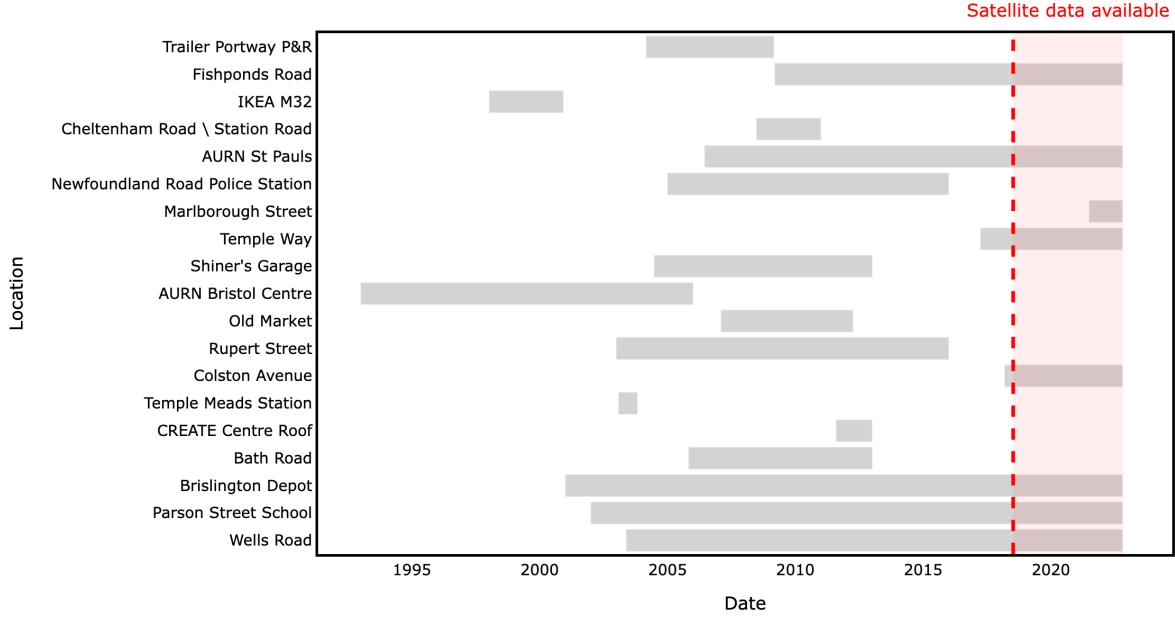


Figure 1: Timeline showing when each sensor in Bristol was active.

### 3.2 Additional Features

Along with the satellite data, we also include various other sources of data in our models, consisting of meteorological data, road traffic related data and the latitude and longitudes of the locations for which to predict.

#### 3.2.1 Satellite Data

For the satellite-based air quality measurements, data from the European Space Agency's Sentinel-5 Precursor satellite was used, which provides worldwide air quality measurements from 2018 - present. Specifically, we use the 'tropospheric column density' of NO<sub>2</sub> - a measure of the concentration of the gas averaged across the lowest 10km of the Earth's atmosphere, and the 'absorbing aerosol index' - a measure of the number of aerosols in the atmosphere, such as smoke and soot. These are recorded with a spatial resolution of 5.5 x 3.5 km and a temporal resolution of roughly 24 hours.

Due to their differing temporal frequencies, when combining the satellite and ground-based data, the satellite data are treated as static for each hourly timestep until the next measurement 24 hours later. While this approach may not capture short-term variations, it allows for the

incorporation of data from a wider spatial area and provides a more complete picture of the air quality situation.

### 3.2.2 Meteorological Data

Hourly meteorological data was acquired from the ERA5-Land hourly dataset [23], consisting of air temperature, wind speed, wind direction, relative humidity, air pressure, vapour pressure deficit, dew point and percentage cloud cover at a spatial resolution of 5km.

### 3.2.3 Road Data

Since high resolution traffic flow data is not widely available in Bristol and London, we can instead provide data on the proximity of each location to the nearest A-road or motorway. This should enable a model to better capture the impact of traffic-related pollutants such as NO<sub>2</sub>. To calculate this feature, the OS Open Roads [24] dataset of all the roads in the UK is used, along with the latitudes and longitudes of each location. A value is calculated for each sensor location, indicating the distance to the nearest A-road or motorway to the nearest metre.

## 3.3 Data Pre-processing

All inputs to the model were normalised using standardisation, defined as follows:

$$x_{std} = \frac{x - \mu}{\sigma} \quad (1)$$

where  $x_{std}$  represents the standardised value,  $x$  is the original value,  $\mu$  is the mean of the feature, and  $\sigma$  is the standard deviation of the feature. This transforms all the inputs to have a mean of 0 and a standard deviation of 1, ensuring that no single feature is prioritised unfairly.

## 3.4 Data Exploration

After extracting and cleaning the ground-based data, we began to look for trends in the air quality measurements.

It was found that air quality across Bristol is highly dependent on the hour of the day, as can be seen in Figure 2a. It can be seen from this plot that, while the effect is more prominent in some locations than others, the time of day has a clear impact on the concentration of NO<sub>2</sub> in Bristol, with a peak at around 7-9am and a second, more gradual peak between 5-7pm. It is also worth noting that this second peak occurs slightly later for locations with lower overall NO<sub>2</sub> levels. This is likely due to the pollution caused by commuting and activity relating to people's sleep schedules. The effect of commuting can also be seen when looking at NO<sub>2</sub> levels for each day of the week, shown in Figure 2b, with the majority of locations experiencing a

gradual increase in NO<sub>2</sub> levels during the week, with a peak on Friday before dropping back down again on the weekend. To improve the prediction accuracy of NO<sub>2</sub> levels, it may be useful to consider incorporating the day of the week as a predictor variable.

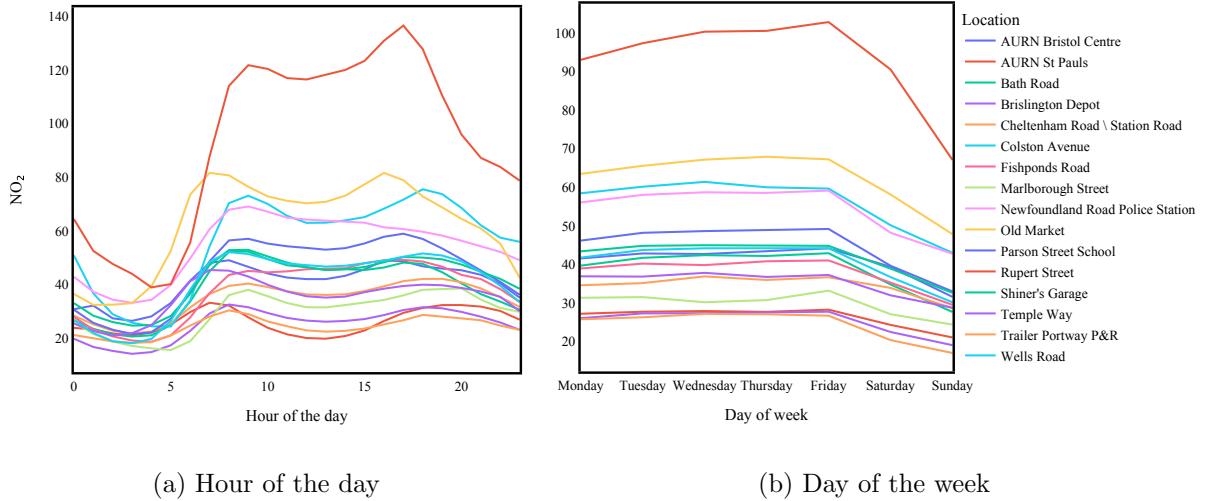


Figure 2: Average NO<sub>2</sub> levels ( $\mu\text{g}/\text{m}^3$ ) for each hour of the day and day of the week in Bristol

It was found that the ground-based and satellite NO<sub>2</sub> readings have a weak, positive correlation, with an average  $r$  value of 0.204. This value does vary significantly between locations, however, with a maximum  $r$  value of 0.503 and a minimum of 0.134. This relatively weak correlation is likely due to local variations in temperature, pressure and air flow caused by activity in the city, therefore it is essential that a model to predict surface level NO<sub>2</sub> should take as many of these variables as input as possible in order to fully understand these patterns.

Since the NO<sub>2</sub> levels vary considerably between locations, an effective model should be able to predict these variations from the input data. From Figure 3, it can be seen that, for the majority of features, there is little to no variation in input data between sensor locations. This can make the task of predicting NO<sub>2</sub> concentrations more challenging, as, if the input data is not diverse enough, the model may not be able to capture the nuances in the relationships between the input variables and the NO<sub>2</sub> levels, which can lead to inaccurate predictions.

### 3.4.1 Analysis of data suitability for transfer learning

Deep learning techniques typically require large amounts of training data to be effective, however, in the case of Bristol, the availability of air quality data is limited, with only 8 active sensors recording NO<sub>2</sub> levels in the same time frame that satellite air quality data is available. This poses a significant challenge for accurately predicting air quality levels in the city. One potential solution to this problem is transfer learning, a widely used technique in machine learning that can help improve the accuracy of predictions in domains where data is scarce. In this paper, we propose to use air quality data from London, with its air quality sensing network consisting of 112 active NO<sub>2</sub> sensors, to improve the air quality predictions in Bristol. By leveraging the

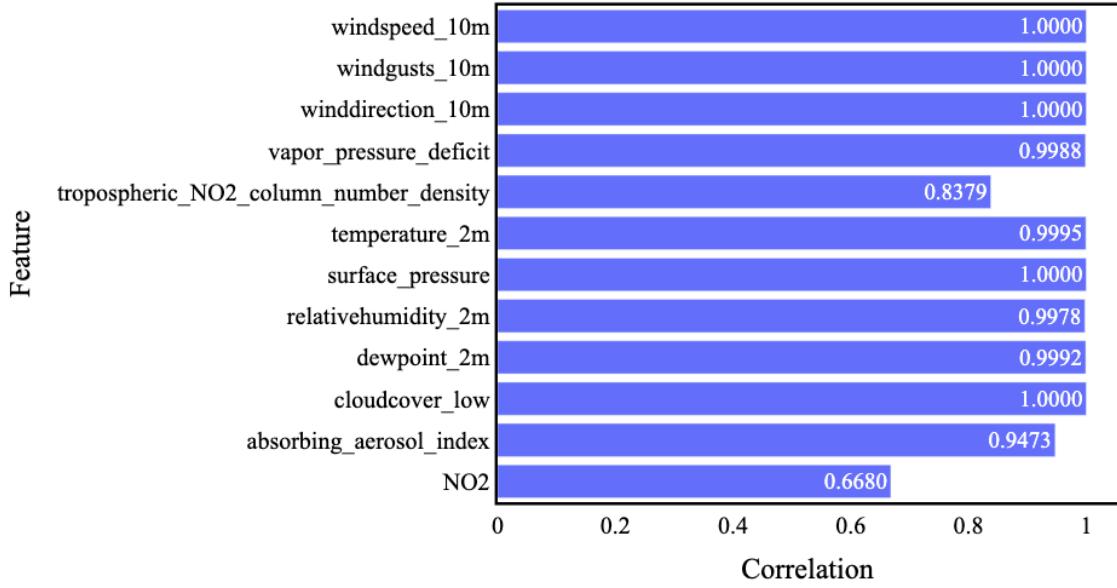


Figure 3: Bar chart showing the average correlation across different sensor locations for each input variable and  $\text{NO}_2$  values.

large amount of air quality data available in London, we can potentially improve the accuracy of air quality predictions in Bristol despite the limited availability of data in the city. The active air quality monitors in both Bristol and London are shown in Figure 4.

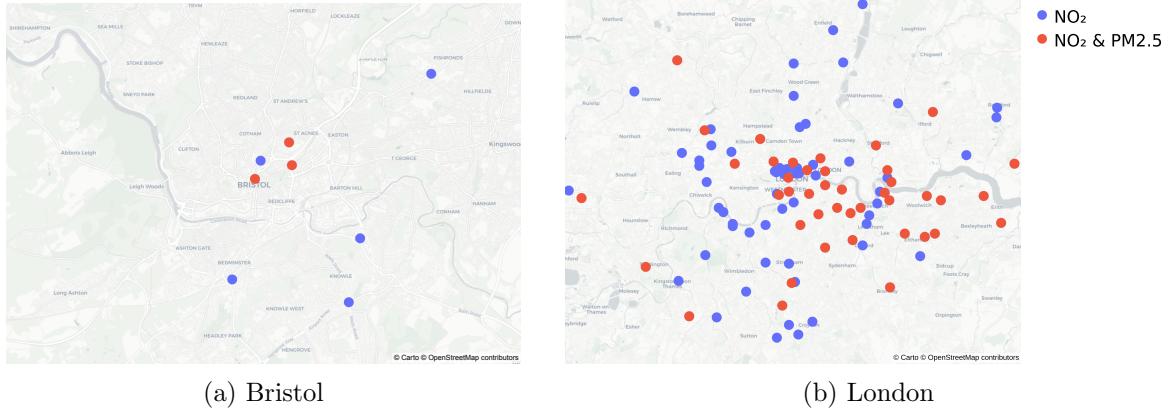


Figure 4: Maps showing locations of active  $\text{NO}_2$  and  $\text{PM}_{2.5}$  sensors in Bristol and London

For transfer learning to be effective, the input data for London must be of the same type and ideally share some characteristics with the Bristol data. From Figure 5a we can see that the two locations exhibit very similar daily patterns, with the  $\text{NO}_2$  levels in London peaking 1-2h sooner in the morning than Bristol, and with a slightly later, more gradual second peak. In Figure 5b, it can again be seen that the two cities follow a similar pattern over the days of the week, with Bristol recording around 30% higher concentrations of  $\text{NO}_2$ .

Looking at Figure 2a and Figure 2b, we can see that, while Bristol records much higher  $\text{NO}_2$  values on average, both London and Bristol share a clear pattern over the course of each day and

week. To enable a model to capture these relationships effectively and to increase the stability of predictions over unseen future times, we can provide multiple different time inputs instead of a linearly increasing value. Here, we provide a *time of day* value, which increases from 0 - 1 over the course of 24 hours; a *day of the week* value, which increases from 0 - 1 from Monday to Sunday; and a *week of the year* value, increasing from 0 - 1 over each year.

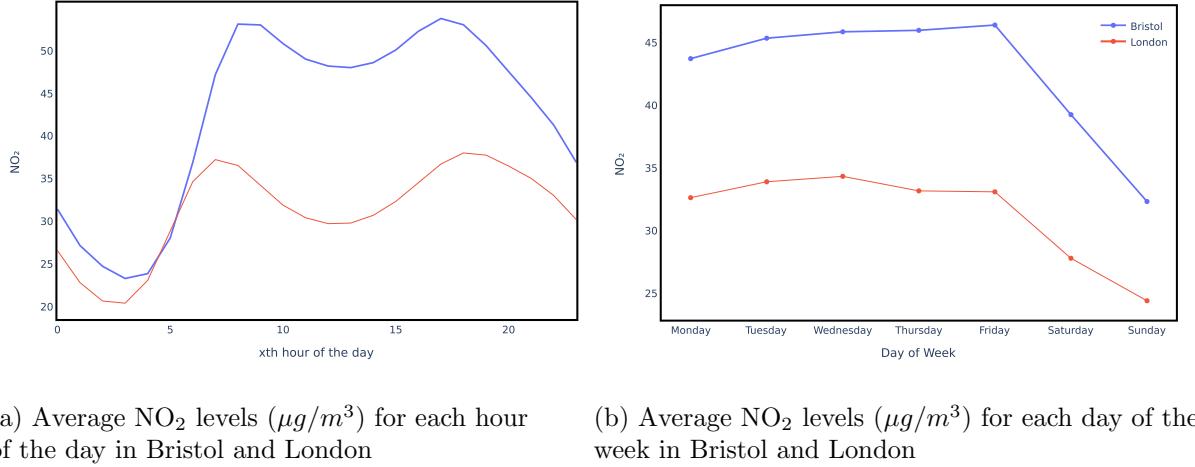


Figure 5: NO<sub>2</sub> levels in Bristol and London

## 4 Models

In order to develop an accurate and reliable model for predicting NO<sub>2</sub> levels in Bristol, we will conduct an evaluation of three models proven to be effective in similar tasks. This section outlines our methodology for comparing the models and incorporates transfer learning with data from London to improve our predictions. By evaluating the performance of each model on unseen locations and assessing the impact of transfer learning, we aim to develop a new model that addresses any issues identified during the evaluation and transfer learning process.

The model that performs best out of the three will then be taken forward and its features and predictions analysed in more detail, including assessing the effectiveness of transfer learning with data from London in improving the predictions on Bristol locations. The results from this model will then act as a performance baseline for NO<sub>2</sub> predictions, allowing us to formulate a new graph-based model, aimed at addressing any inherent limitations. Due to its complexity and performance, this model will be discussed at a greater depth than those that precede it.

Each model will take as input all the variables described in section 3, and will output hourly NO<sub>2</sub> values for the unseen location in  $\mu\text{g}/\text{m}^3$ . Each model will be tested on one location at a time, having trained on all the remaining locations, and a single error metric will be produced by averaging across all locations. Testing in this way ensures that each model is evaluated on previously unseen locations, providing a robust evaluation of their ability to act as a ‘virtual

sensor' for the Bristol area. The input features for all models are:

**Satellite Features:** Tropospheric NO<sub>2</sub> column number density, absorbing aerosol index;

**Meteorological Features:** Wind speed, wind gust speed, wind direction, vapour pressure deficit, temperature, surface pressure, relative humidity, dew point, cloud cover percentage;

**Time-based Features:** Day of the week, week of the year and time of day.

Model performance will be assessed using the following 3 metrics:

1. Root Mean Squared Error (RMSE): RMSE is a commonly used metric to measure the difference between predicted values and actual values. It calculates the square root of the average of the squared differences between predicted and actual values. The formula for RMSE is:

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

2. Normalised Root Mean Squared Error (NRMSE): NRMSE is a variation of RMSE that takes into account the scale of the observed values. It is calculated by dividing the RMSE by the range of the observed values. The formula for NRMSE is:

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

where  $y_i$  is the observed value,  $\hat{y}_i$  is the predicted value and  $n$  is the number of observations.

3. Gradient RMSE (Grad-RMSE): We define the metric Grad-RMSE to measure the difference between the predicted and the actual gradients at each timestep. This metric should give a better idea of how well the model is able to predict spikes in NO<sub>2</sub> values, which can be an important factor when dealing with air quality. The formula for Grad-RMSE is:

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

where  $\Delta y_i = y_i - y_{i-1}$  is the difference between adjacent observed values,  $\Delta \hat{y}_i = \hat{y}_i - \hat{y}_{i-1}$  is the difference between adjacent predicted values, and  $n$  is the number of observations.

The following 3 models were tested:

- an XGBoost model [25] with 100 decision trees;
- a multilayer perceptron model (MLP) with 2 fully connected layers, a dropout layer with a rate of 0.5, and 2 more fully connected layers;

- a convolutional neural network (CNN) model with 2 convolutional layers, a dropout layer with a rate of 0.5, and 2 fully connected layers.

The performance of these models is shown in Table 1. From these results, it can be seen that the CNN model achieved the lowest RMSE and NRMSE, but the XGBoost model achieved a marginally better Grad-RMSE, indicating that, while the XGBoost model may be slightly stronger in predicting the change in NO<sub>2</sub> concentration, the CNN model is more accurate overall.

	RMSE	NRMSE	Grad-RMSE
MLP	27.482	0.876	10.812
CNN	<b>21.133</b>	<b>0.672</b>	9.741
XGBoost	22.773	0.721	<b>9.583</b>

Table 1: Performance of 3 initial models in predicting NO<sub>2</sub> values for unseen locations in Bristol, averaged across all locations.

#### 4.1 Convolutional Neural Network Model

The CNN model follows the same architecture as that used in the study by Ghahremanloo et al. [6], which achieved an NRMSE of 0.35 when predicting NO<sub>2</sub> concentrations at a resolution of 7 × 5.5km over Texas, leveraging several extra input variables such as enhanced vegetation index and road density. A schematic view of the model architecture is shown in Figure 6.

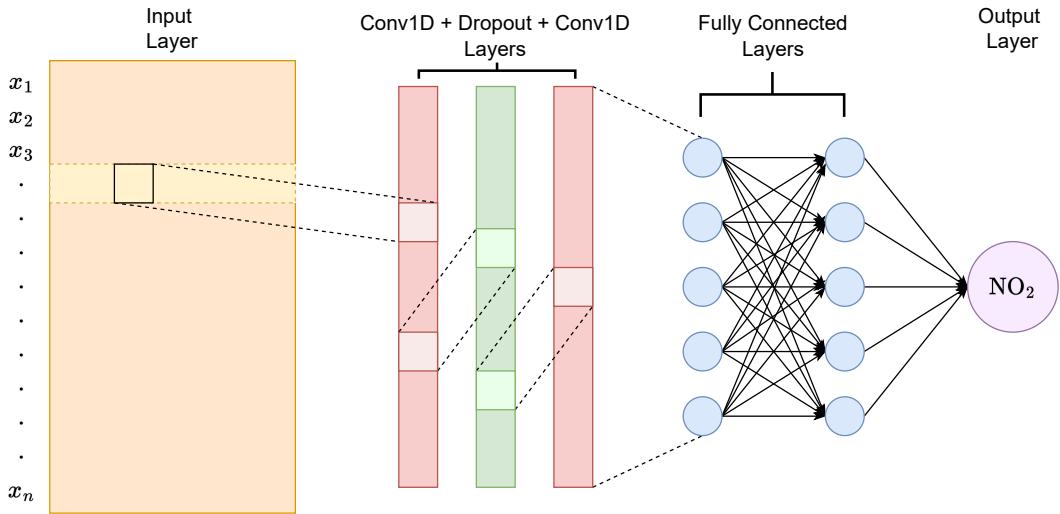


Figure 6: Schematic view of the architecture of the CNN model

This model achieved an NRMSE of 0.67 when predicting NO<sub>2</sub> in Bristol, a decrease in performance which is likely due to reduction in the number of input variables, the increased complexity of predicting at sensor locations in a relatively small area and the lack of variation in input data this entails. Nonetheless, this performance suggests that the CNN model is still able to effectively learn the underlying patterns in the data and make accurate predictions of NO<sub>2</sub> concentrations.

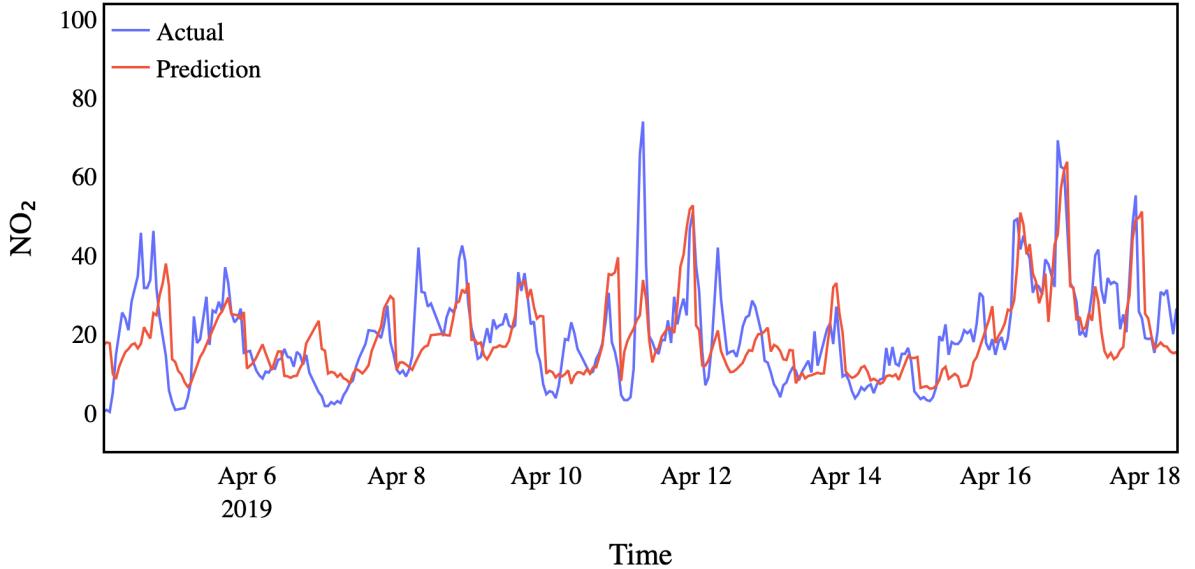


Figure 7: A sample of NO<sub>2</sub> predictions from the CNN model for Well’s Road, Bristol.

Looking at a sample of the NO<sub>2</sub> predictions for one location in Bristol (shown in Figure 7), it is evident that the accuracy of the predictions fluctuates significantly over time, with some spikes being captured almost perfectly, whereas predictions at other time periods bear little relation to the actual NO<sub>2</sub> values. The accuracy of the model also varies greatly between sensor locations, the map in Figure 8 shows that the model performs considerably worse on the two locations in more densely populated areas, suggesting that it may be failing to differentiate between locations from its previous NO<sub>2</sub> levels and instead relying more heavily on satellite and weather data.

This issue is highlighted when we look at the variation in the predictions for each location compared to the variation of the actual NO<sub>2</sub> values, shown in Figure 9a. While the CNN model does predict a large range of values for different locations, it seems to fail at predicting the variations at the right times. Moreover, it is evident from the mean of predictions across all locations that the model is often conservative in its predictions, consistently under-predicting spikes in NO<sub>2</sub> and over-predicting dips.

#### 4.1.1 CNN transfer learning

Transfer learning for the CNN model was achieved by first training the model on the London data, then freezing the weights of the first 3 layers, and then fine-tuning the model on the Bristol data. This allows the features extracted by the convolutional layers on the London data to be carried over for use on the Bristol data. Transfer learning in this way was found to improve the NRMSE and RMSE by 10.34% and 10.19% respectively, however the Grad-RMSE got worse by 4.82%. These results suggest that transfer learning can be an effective technique for improving the performance of the CNN model on new data. The significant improvements in NRMSE and RMSE show that the features extracted from the London data were indeed relevant to the Bristol data. However, the slight increase in Grad-RMSE indicates that the transferred model struggles

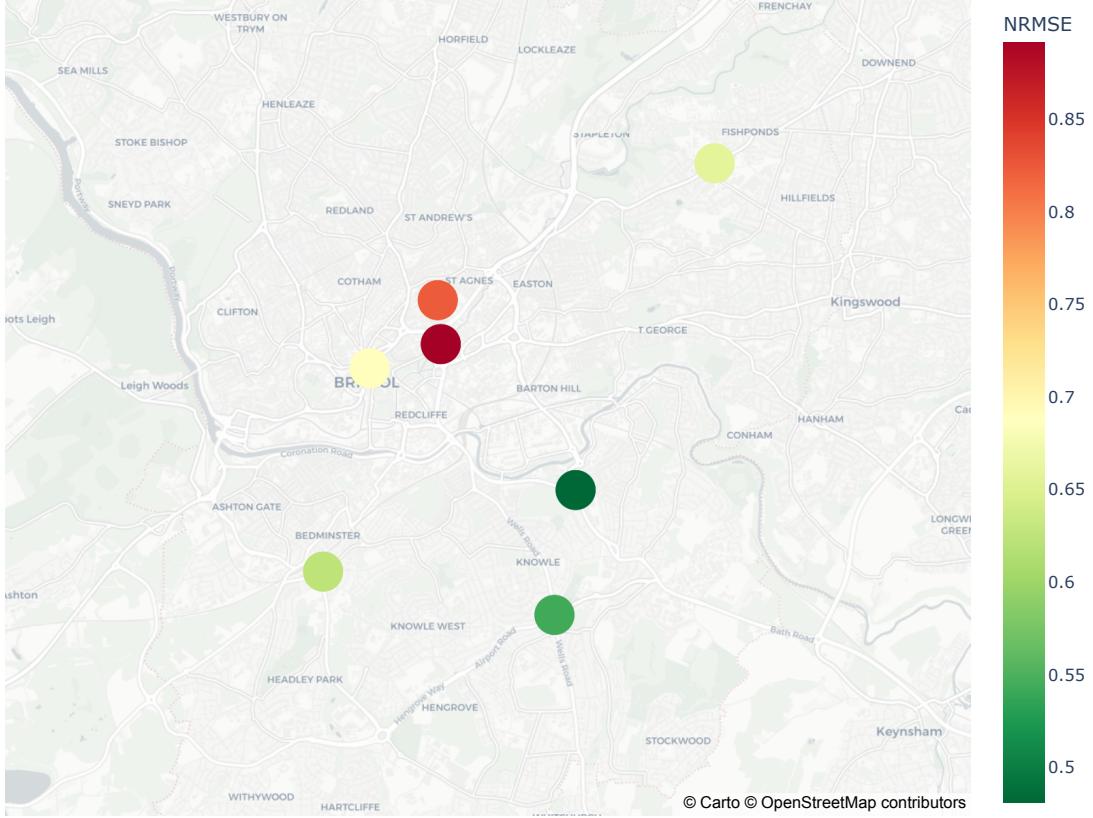
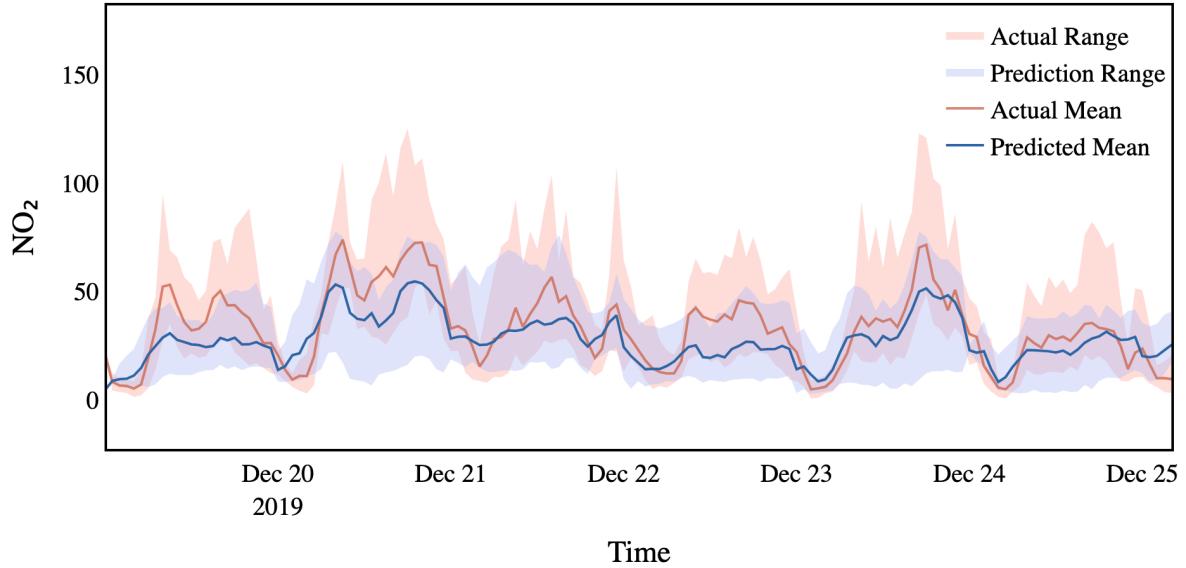


Figure 8: NRMSE for the CNN model predictions for each sensor location in Bristol.

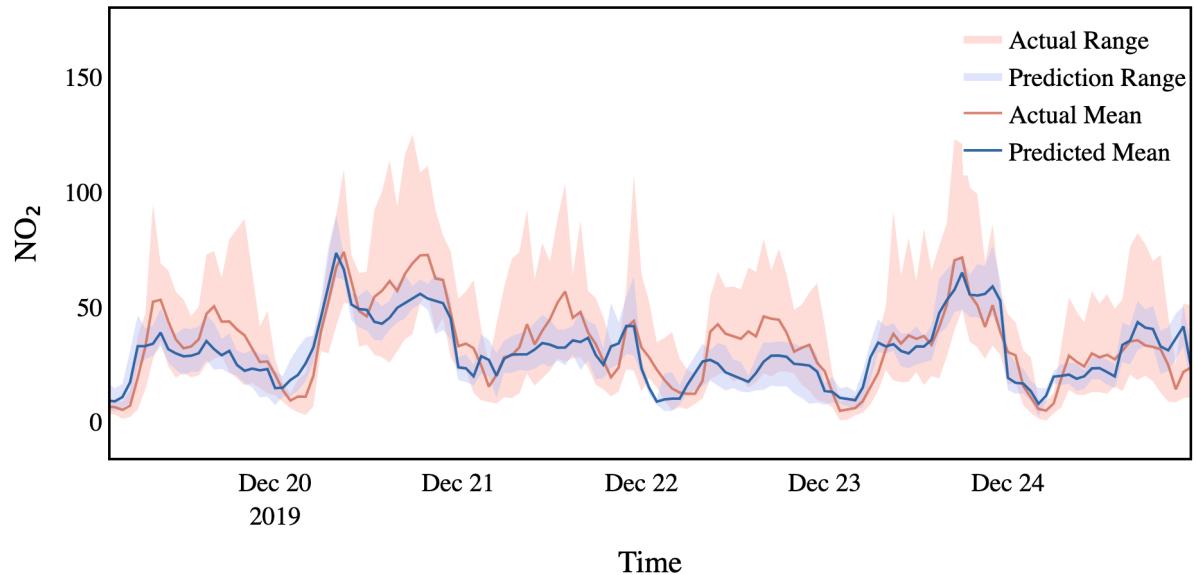
to capture the changes in  $\text{NO}_2$  values over time. This suggests that the features extracted from the London data may not have been as effective in capturing the temporal patterns of  $\text{NO}_2$  levels in the Bristol data.

To gain a deeper understanding of the impact that pre-training on the London data has the model, we can look at the variations of the  $\text{NO}_2$  predictions between different sensor locations. From Figure 9a and Figure 9b, it can be seen that the model trained on just the Bristol data predicts a much greater range of values between different sensor locations, often predicting  $\text{NO}_2$  values at a level much below the true values. This is in contrast to the transferred model, which appears to predict a much smaller range, tending not to predict outside of the true range of values. While it is important for a model to be able to predict extremes in  $\text{NO}_2$  concentrations, the model trained on just the Bristol data does not seem to be able to capture spikes effectively, instead predicting large variations across locations with little resemblance to the true values.

In this case, the transferred model may be more useful in producing meaningful predictions, due to its tendency to stay within the true  $\text{NO}_2$  values and correctly capturing when  $\text{NO}_2$  levels will spike, even if it does not predict with the same magnitude. While the two plots shown here only represent a small portion of the data, this trend was observed throughout the dataset.



(a) A 6 day sample of the range and average of predicted and actual NO<sub>2</sub> values across all locations in Bristol for the CNN model.



(b) A 6 day sample of the range and average of predicted and actual NO<sub>2</sub> values across all locations in Bristol for the transferred CNN model.

Figure 9: Comparison of the range and average of predicted and actual NO<sub>2</sub> values across all locations in Bristol for the CNN model and the transferred CNN model.

## 4.2 Autoregressive GraphSAGE Model

In this section, we introduce a new model to overcome the limitations of the CNN model in capturing the variations between sensor locations. The proposed model is based on the GraphSAGE algorithm, which was first introduced by Hamilton et al. in 2017 [26]. We will first explain the advantages that GraphSAGE offers and how its ‘sample and aggregate’ algorithm is executed, before describing how this model will be applied to the task of predicting NO<sub>2</sub> values for unseen sensor locations by incorporating an autoregressive time component. Two methods for improving the model by using the London data will then be discussed and their performance analysed, assessing the extent to which they improve the accuracy of the model. By implementing these two methods, we aim to demonstrate that pre-training on a larger and dataset can improve the accuracy of the GraphSAGE model in predicting NO<sub>2</sub> values across the range of Bristol locations.

### 4.2.1 GraphSAGE

Unlike traditional graph neural networks, such as GCNs, GraphSAGE is an inductive model that learns node representations by aggregating information from its local neighbourhood of connected nodes. In our case, each node represents a sensor location and edges are created between node pairs that are within a distance of  $d$ , where  $d$  is the maximum distance between the furthest-away node and its nearest neighbour. This ensures that no nodes are isolated, allowing for a more efficient use of the graph.

GraphSAGE’s inductive property provides several benefits over typical GNNs, firstly, it enables predicting on unseen nodes, which will allow us to create ‘virtual sensors’ without the node being present in training. This is useful as it allows the model to generalise to new and unseen data without the need for additional training. Secondly, since GraphSAGE does not require every node to be present at each time step, it eliminates the need for data imputation before training the model. This will result in a more generalisable model, enabling it to be used in areas where air quality data may be inconsistent. For large networks of air quality sensors, GraphSAGE has another advantage in that it aggregates data from each node’s local neighbourhood of sensors, allowing it to better capture geographical variations in air quality.

The GraphSAGE algorithm for sampling and aggregating information from its local neighbourhood is described as follows:

Given a graph  $G = (V, E)$ , where  $V$  is the set of nodes and  $E$  is the set of edges in the graph, node features are denoted as  $x_v, \forall v \in V$ . The model uses  $K$  aggregation functions, denoted as  $\text{AGGREGATE}_k, x_v, \forall k \in \{1, \dots, K\}$ , to aggregate information from node neighbours, and the corresponding weight matrices  $W^k, \forall k \in \{1, \dots, K\}$ , are used to propagate information between different layers.

To generate node embeddings, the GraphSAGE model performs the following procedures:

1. Each node  $v$  aggregates the representations of nodes in its immediate neighbourhood  $N(v)$  to the neighbourhood vector  $h_{N(v)}^k$ :

$$h_{N(v)}^k = \text{AGGREGATE}_k \left( h_u^k, \forall u \in N(v) \right),$$

where the initial representations of nodes are the input node features  $h_v^0 = x_v, \forall v \in V$ .

2. The node's current representation  $h^{k-1}$  is concatenated with its aggregated neighbourhood vector  $h_{N(v)}^{k-1}$ , which is later fed into a fully connected layer with a nonlinear activation function  $\sigma$ , and then used for the representations for the next step such that:

$$h_k = \sigma \left( W^k \cdot \text{CONCAT} \left( h^{k-1}, h_{N(v)}^k \right) \right).$$

The  $\text{CONCAT}(\cdot)$  function concatenates inputs along a specified dimension. [26], and the node representations become:

$$h_v^k \leftarrow \sigma \left( W^k \cdot \text{MEAN} \left( \left\{ h_v^{k-1} \right\} \cup \left\{ h_u^{k-1} \right\}, \forall u \in N(v) \right) \right),$$

where  $\text{MEAN}(\cdot)$  computes the mean of inputs along a specified dimension.

3. The final representation output  $z_v = h_v^K, \forall v \in V$  at depth  $K$  is the output of the embedding generation.

When given a new, unseen node to predict on, the model uses the node representations learned in the training process to generate a node embedding for the new node, based on its connected neighbourhood of nodes. This node embedding is then used to aggregate information from its neighbours as well as its own features to produce the final representation, resulting in the NO<sub>2</sub> prediction for that timestep, a simplified example of which is shown in Figure 10.

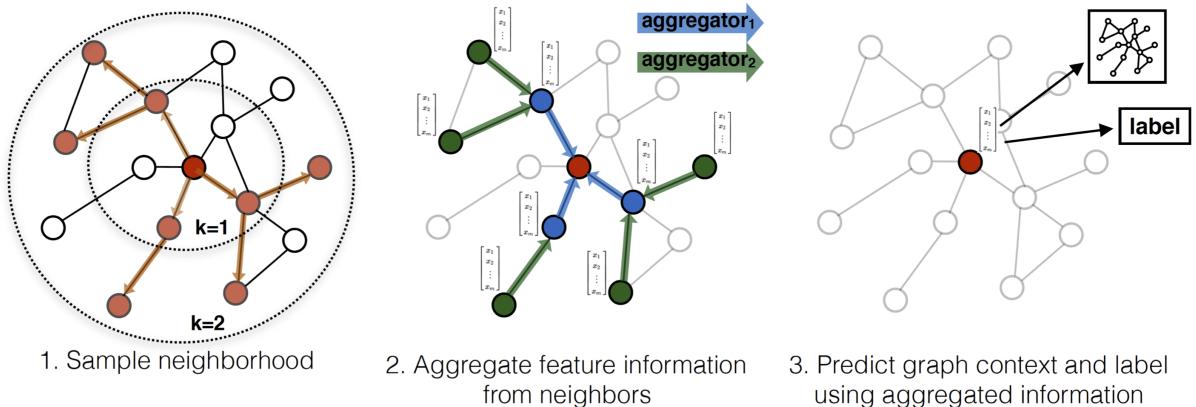


Figure 10: Diagram showing the GraphSAGE sample and aggregation process, taken from the paper: Inductive Representation Learning on Large Graphs - Hamilton et al.. [26]

#### 4.2.2 Training process

GraphSAGE models are widely used for modelling relationships on large graphs, having been used to power recommender systems for companies like Uber and Pinterest [27], [28]. Previous approaches for modelling air quality using GNNs, such as [9], involve feeding in a list of graphs, representing the sensor network at each timestep, and then passing the learned attributes into an RNN or LSTM to capture how these attributes change over time. GraphSAGE, however, takes as input a single large graph and randomly samples from each nodes neighbourhood as it learns aggregation functions. To form a single graph, we take our graphs for each timestep (every hour) and combine them into one graph, as demonstrated in Figure 11. This results in one large graph with many disconnected subgraphs, one for each timestep. Since the GraphSAGE algorithm works by sampling from a node’s local neighbourhood of connected nodes, feeding in the data all at once in this way does not allow the model to see other timesteps in the future or the past, as the graphs at each timestep are not connected to each other.

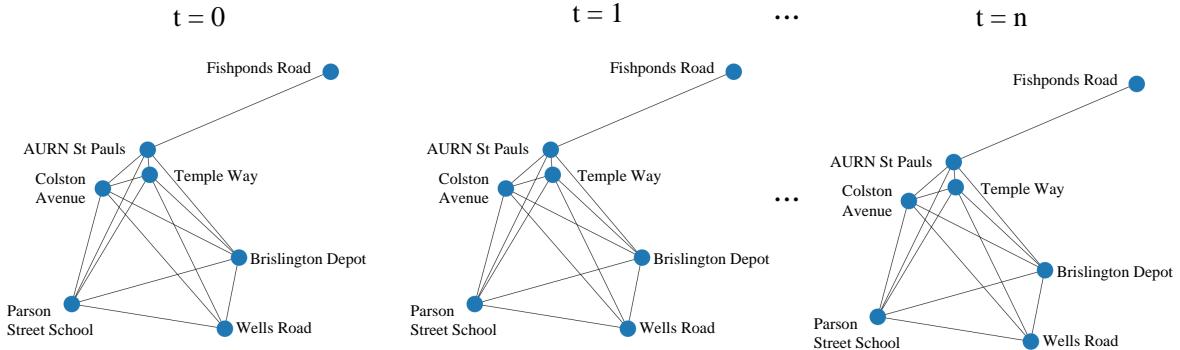


Figure 11: An example of how the sensor networks in Bristol are combined to form a single large graph for use with GraphSAGE.

In order to address this issue of not having a continuous representation of time in the graph, we include the predicted NO<sub>2</sub> value from a node’s previous timestep as an input for the current prediction in a process called autoregression. Autoregression is a technique used to model time series data, where each data point is predicted based on the previous predicted values. During training, we include the previous timestep’s actual NO<sub>2</sub> value at each node as a feature for that node. When nodes at a certain timestep are missing, the last recorded value for that time of day is used in its place. This should not greatly affect our results due to the high correlation the NO<sub>2</sub> values have with time. This enables the model to learn how to aggregate the satellite data, meteorological data and previous NO<sub>2</sub> values for each node and its neighbours. The use of autoregression is particularly effective for modeling time series data that exhibit a high degree of autocorrelation, as is the case with the NO<sub>2</sub> concentration data. Autocorrelation is a measure of the degree to which a data point is correlated with its preceding data points, and can be visualised using an autocorrelation plot (see Figure 12). As can be observed from the plot, there is a significant degree of correlation between the NO<sub>2</sub> concentration at each timestep and the NO<sub>2</sub> concentration at the next timestep.

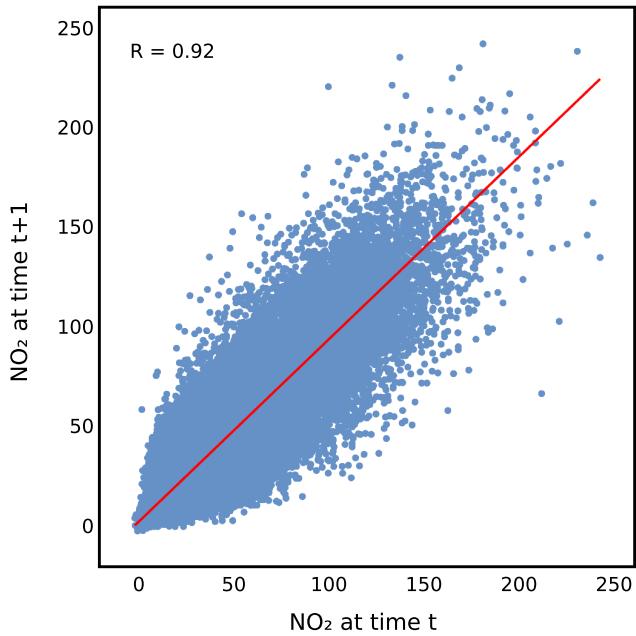


Figure 12:  $\text{NO}_2$  values at time  $t$  vs  $\text{NO}_2$  values at time  $t + 1$ .

By incorporating the autoregressive feature in our model, we are able to capture this correlation and make better predictions for the  $\text{NO}_2$  concentration at each node. This leads to a more accurate representation of time in our model and improves its ability to make predictions on future timesteps.

To predict on an unseen node, we must first initialise the node with a value for the  $\text{NO}_2$  at the previous timestep. During development of the model, this was achieved by including the actual  $\text{NO}_2$  value for the first timestep, however in reality this initial sample could be provided by an air quality sampling scheme such as the Breathe London scheme, which involves children wearing portable air quality monitors on their backpacks [29], or LocalAir, an e-scooter based air quality monitoring scheme in Bristol [30]. These methods of sampling are designed to be cheap and versatile in their applications so could be used in countries lacking infrastructure to provide a baseline  $\text{NO}_2$  reading for a new location. If no such sample is available, it would also be possible to simply provide an estimate for the  $\text{NO}_2$  concentration at a particular location, as the model only has access to a single previous value, forcing it to focus on the change in  $\text{NO}_2$  at each timestep.

#### 4.2.3 Model parameter selection

To ensure the best performance on the data, we compare the performance of four different node aggregator functions, mean, max pooling, mean pooling and attentional aggregator, as defined in the StellarGraph python library [31].

Due to the time and resources required to train and predict with this model, the different

aggregator functions were tested on two randomly selected locations from the Bristol dataset (Brislington Depot and Parson Street School) instead of retraining to predict on each location in the dataset. The results for these two locations were then averaged, the results of which are shown in Table 2, indicating that the mean aggregator function performed the best on the Bristol dataset. For this reason, mean was chosen as the aggregator function for the model.

	NRMSE	RMSE	Grad-RMSE
Mean	<b>0.528</b>	<b>10.592</b>	<b>10.379</b>
MaxPool	0.595	11.683	11.776
MeanPool	0.574	11.278	11.598
Attention	0.530	10.619	10.794

Table 2: NRMSE, RMSE and Grad-RMSE for different aggregator functions when averaged across two locations in the Bristol dataset.

To improve the model’s ability to generalise to unseen data, we include a dropout layer before the activation layer, parameters for which were experimented with, as shown in Figure 13. From these results, a dropout rate of 0.5 was chosen.

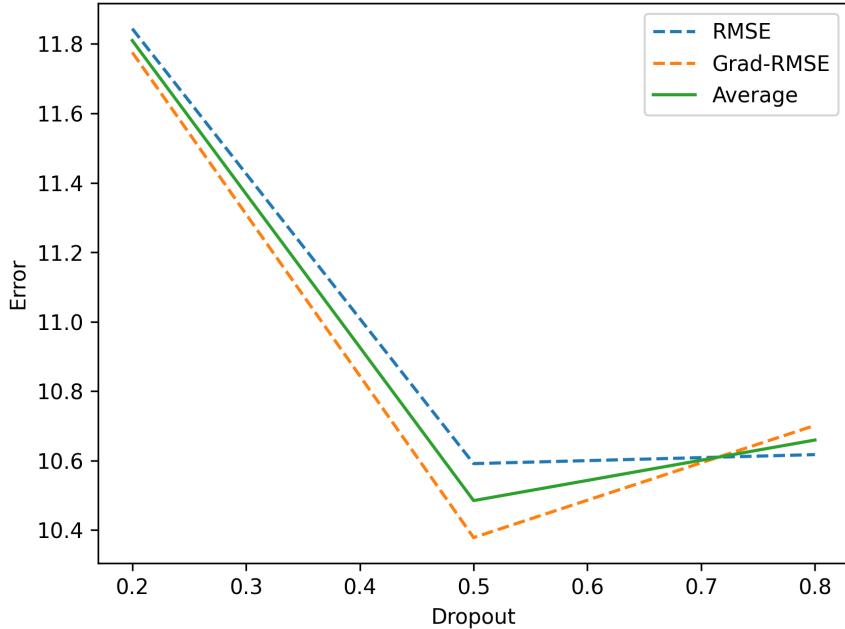


Figure 13: RMSE and Grad-RMSE performance of the model using different dropout values, when averaged across 2 locations in the Bristol dataset.

Other parameters for the model include the number of hops away from each node to sample from, the maximum number of nodes to sample at each hop and the number of neurons to use when aggregating each node and its neighbours. Since the maximum number of hops possible from any node in the Bristol graph is 2, it was decided that we would perform 2 aggregations, sampling nodes at one hop, then two hops from each node. At each of these steps, a maximum

of 3 and 5 nodes would be sampled respectively. Other parameters for the model such as the number of neurons for aggregation and the learning rate were selected by trial and error.

Due to the nature of the input graph for the GraphSAGE model, we explore two different methods for allowing the model to learn on both Bristol and London data. The first method involves combining both the Bristol data and the London data on the same graph, so that it can train on all of the data at each epoch, as shown in Figure 14. The second method is a more typical transfer learning approach, involving training a model on the London data first, then using the weights from this model to initialise a model for training on the Bristol data.

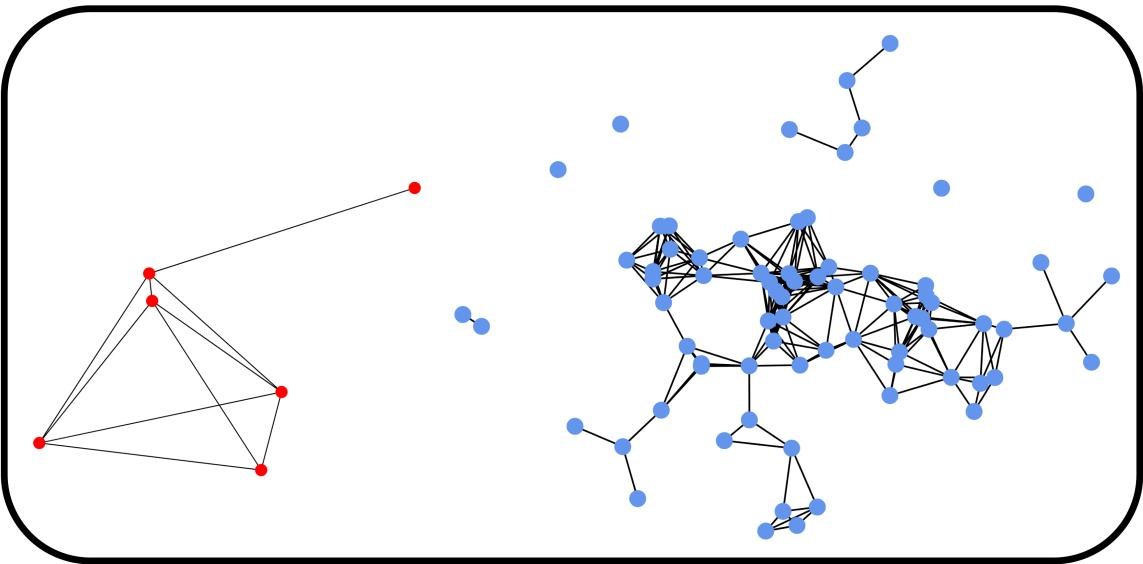


Figure 14: Example of the combined graph for a single timestep, with Bristol locations shown in red, London in blue. Note that the actual input to the model would be one such graph for every timestep, as shown in Figure 11.

Looking at an example of the predictions from a model trained on the combined graph, shown in Figure 15, we can see that the large imbalance in the amount of data between the two cities may have caused the model to overfit on the London data. To investigate this further, we test the models performance when including varying amounts of London nodes in its training data, starting from zero and increasing the number of London nodes to find the optimum number so as not to ‘swamp’ the Bristol data. The selection of London nodes to include was performed by steadily increasing the radius of a circle around the centroid of all the London locations, until the number of nodes inside the circle reached the desired number for which to test. We can see from Figure 16 that the model performed best when trained on just Bristol nodes. Due to its poor performance, this method of incorporating the London data to predict on unseen sensor locations in Bristol will not be included in further comparisons.

The second method of transfer learning, involving first training a model on just the London data, then fine-tuning this model to predict NO<sub>2</sub> in Bristol, gave much better performance. The two models are compared in Table 3, where it can be seen that the transferred model resulted in improved performance in all three metrics

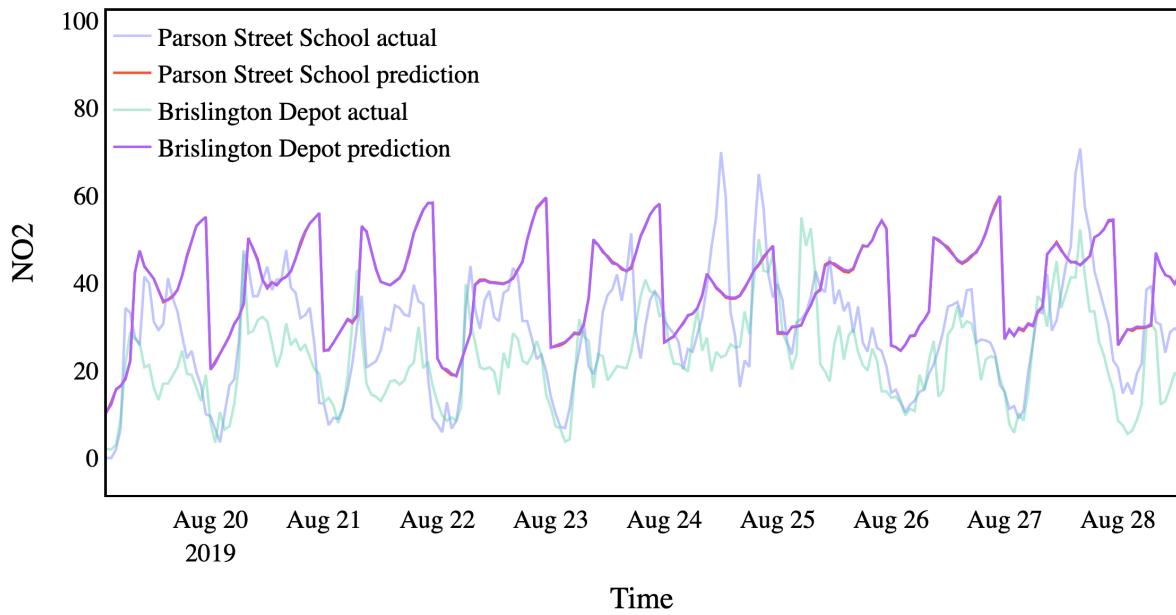


Figure 15: NO<sub>2</sub> predictions from a GraphSAGE model trained on both Bristol and London data at once. Note that the predictions for the two locations are almost identical, indicating that the model has overfitted the London data.

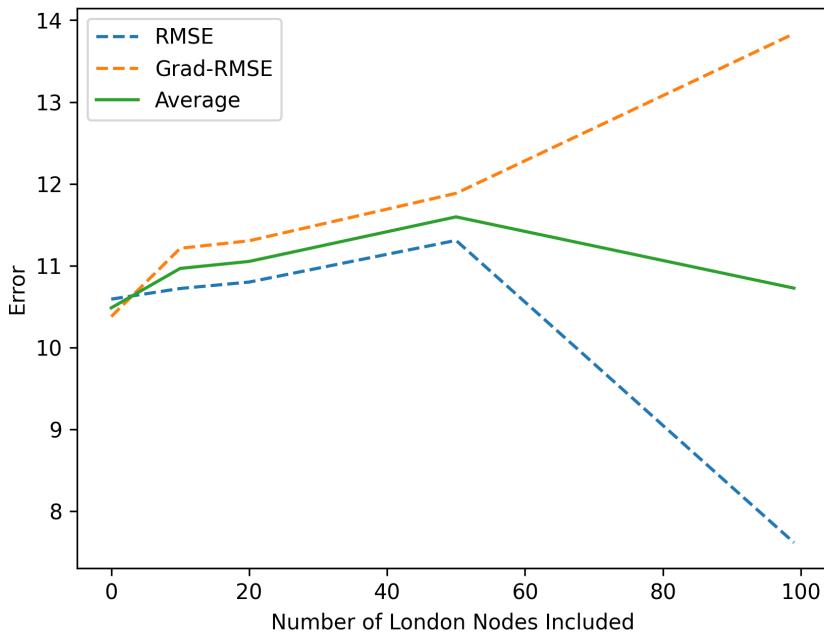


Figure 16: Performance of GraphSAGE model when trained on varying amounts of London data concurrently with the Bristol data.

	NRMSE	RMSE	Grad-RMSE
GraphSAGE	0.526	17.016	9.426
Transferred GraphSAGE	0.481	15.623	6.354
Percentage Improvement	8.576	8.185	32.593

Table 3: Comparison of NO<sub>2</sub> prediction performance between the GraphSAGE model and the transferred GraphSAGE model

## 5 Results and Discussion

This project aimed to develop a model that accurately predicts NO<sub>2</sub> concentrations at unseen locations in Bristol using satellite air quality and meteorological data.

In order to gauge the feasibility of this task, we first explored the data available in Bristol and London, assessing the relationships between our predictor variables and the ground-based NO<sub>2</sub> readings. We then first tested three basic methods for predicting NO<sub>2</sub> concentrations: an XGBoost model, an MLP model and a CNN model. We found that the CNN model performed best, achieving an average NRMSE of 0.672, and RMSE of 21.133  $\mu\text{g}/\text{m}^3$  and a Grad-RMSE of 9.741  $\mu\text{g}/\text{m}^3$  when tested on unseen sensor locations in Bristol. By using transfer learning with air quality data from London, we were able to reduce the NRMSE of this model by 13.24%, although this led to an increase in the Grad-RMSE of 1.76%. This initial testing gave us insight into the effectiveness of traditional machine learning methods and transfer learning in predicting NO<sub>2</sub> values. However, we also observed limitations of these models, such as the inability to effectively capture temporal patterns and accurate variations between sensor locations.

We then proposed a novel method, based on the GraphSAGE inductive learning framework, to address the limitations of the CNN model. This was achieved by combining all training data into a single large graph and allowing the model to learn temporal patterns through autoregression, meaning it predicts NO<sub>2</sub> values one at a time, feeding the prediction at one timestep as input for the next.

We found that the GraphSAGE model was able to predict changes in NO<sub>2</sub> concentrations in Bristol with a high degree of accuracy, with an NRMSE of 0.526, an RMSE of 17.016  $\mu\text{g}/\text{m}^3$ , and a Grad-RMSE of 9.426  $\mu\text{g}/\text{m}^3$ .

Transfer learning with data from London was shown to be effective in improving NO<sub>2</sub> predictions, by first training a model to predict on data from London, then fine-tuning it to predict on unseen sensor locations in Bristol. We found that transfer learning was able to decrease the error in NO<sub>2</sub> predictions for the GraphSAGE model, achieving an 8.58% reduction in NRMSE and a 32.59% reduction in Grad-RMSE. A summary of all the models' performance can be found in Table 4.

In order to gain a more complete understanding of how effective the transferred GraphSAGE model is in predicting NO<sub>2</sub> values, we can compare its predictions against those produced by

	NRMSE	RMSE	Grad-RMSE
MLP	0.876	27.482	10.812
XGBoost	0.721	22.773	9.583
CNN	0.672	21.133	9.741
Transferred CNN	0.583	18.362	9.912
GraphSAGE	0.526	17.0159	9.426
Transferred GraphSAGE	<b>0.481</b>	<b>15.623</b>	<b>6.354</b>

Table 4: Average error metrics for all models included in the study. Note that these error metrics are averaged across each sensor location in the Bristol dataset.

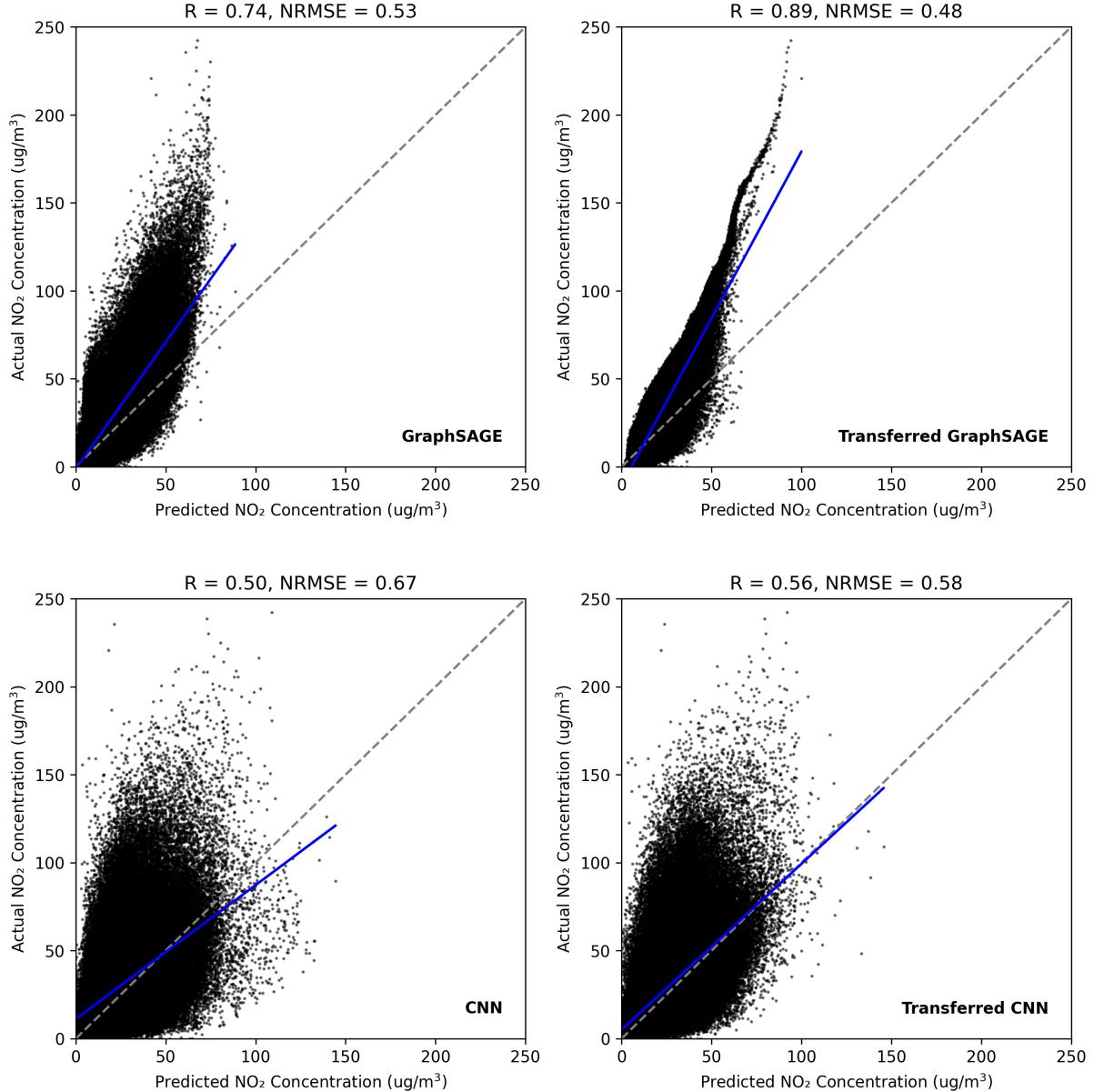


Figure 17: Scatterplots showing the performance of the base GraphSAGE, Transferred GraphSAGE, CNN and Transferred CNN on predicting  $\text{NO}_2$  concentrations across unseen sensor locations in Bristol.

other models analysed thus far. Figure 17 shows us that both CNN models and GraphSAGE models frequently seem to under-predict, however this is more apparent for the GraphSAGE models. It is also clear in both models that transfer learning with the London data increased the correlation between the predictions and the actual values, and that transfer learning was more effective on the GraphSAGE model, as evidenced by the larger increase in  $r$  value and decrease in width of the scatter. This observation suggests that the GraphSAGE model may be better at consolidating information learned from different areas, which would be advantageous if the model were to be used in areas with few NO<sub>2</sub> monitoring stations.

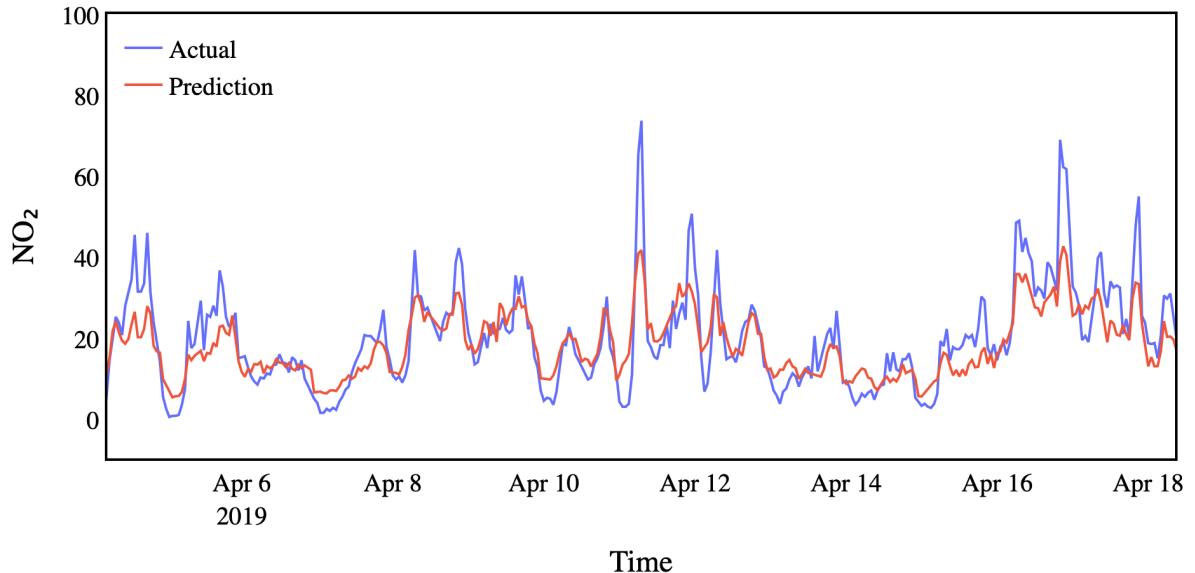
By looking at the sample of predicted versus actual NO<sub>2</sub> values for two locations in Figure 18, we are able to see evidence of this high correlation, as well as its tendency to under-predict higher NO<sub>2</sub> values. This effect is made especially clear in Figure 18b, where the predicted values seem to follow the true values accurately, but are scaled down by some factor.

## 6 Conclusions and Future Directions

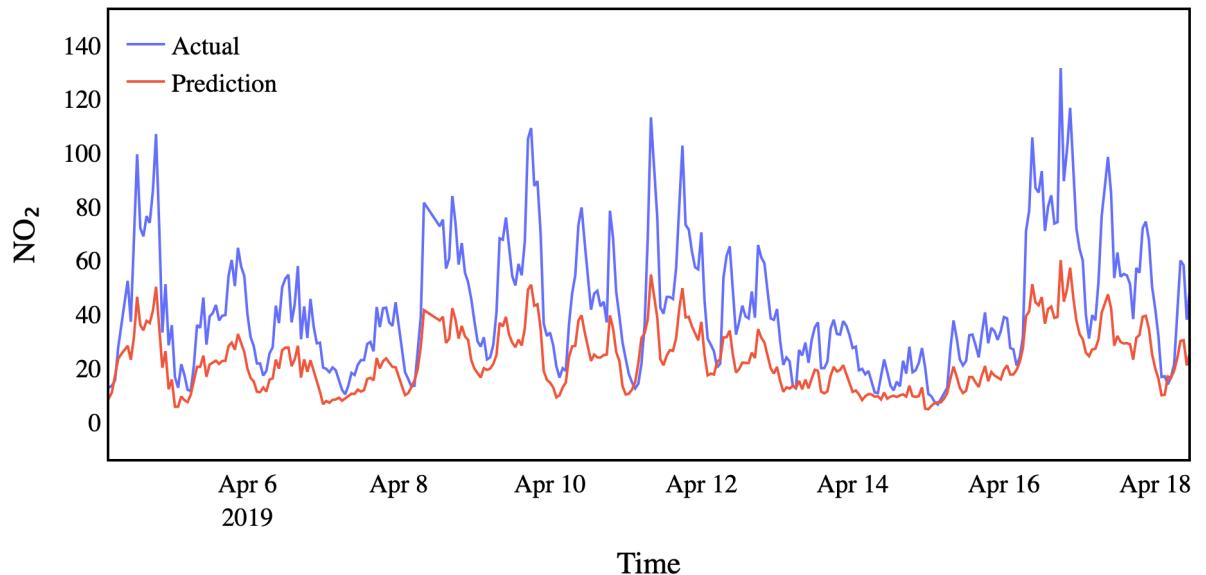
The GraphSAGE model proposed in this study has the potential to be an effective tool for creating virtual sensors in areas with limited air quality monitoring resources, such as developing countries. This is because the model can be trained on a small set of ground-based sensor data and then used to predict NO<sub>2</sub> concentrations locations at unseen locations. The model has several advantages over existing methods in that it can predict with greater accuracy at specific locations, instead of averaging over a large area, enabling more precise pollution monitoring. Furthermore, the graph structure of the model enables it to take into account the NO<sub>2</sub> values of any nearby monitoring stations, improving the accuracy of its predictions and enabling it to adjust to changes in the local landscape that are not reflected in satellite and meteorological data alone.

Additionally, transfer learning was shown to be effective in improving the performance of the model in the target area. By first training on data from London, the model was able to achieve a much lower NRMSE compared to the model trained solely on Bristol data.

However, further testing is needed to assess the accuracy of the model in predicting NO<sub>2</sub> concentrations in more diverse climates. This could involve collecting more data from different regions, possibly including more predictor features in order to help the model distinguish between more diverse environments. In order for this model to be deployed effectively in areas lacking air quality monitoring infrastructure, more research is necessary to optimise transfer learning across multiple distinct areas, enabling a model to learn from more than one other location. In addition, energy consumption and time efficiency of the GraphSAGE model would need to be addressed before deployment in such an area. This is especially relevant as the model is relatively computationally expensive in its current form, requiring the input graph to be recreated at every timestep in order to input the previous timestep's NO<sub>2</sub> prediction as a node attribute.



(a) NO<sub>2</sub> predictions from the transferred GraphSAGE model for Well's Road, Bristol.



(b) NO<sub>2</sub> predictions from the transferred GraphSAGE model for Colston Avenue, Bristol.

Figure 18: Actual and predicted NO<sub>2</sub> values from the transferred GraphSAGE model for Well's Road - a location with typically low NO<sub>2</sub> values, and Colston Avenue - a location with typically high NO<sub>2</sub> values.

In relation to the performance of the model on the data used in this study, more research could be done around optimising the loss function in order to reduce the issue of under-predicting. This could take the form of a function that penalises values below the true value more heavily than those above. More research could also be done in experimenting with different initial sample values, in order to gauge the impact that baseline samples have on the rest of the predictions. Similarly, an investigation into the effect of including a greater number previous predictions as input to the model at each timestep could provide valuable information for improving the performance and resilience of the model.

## References

- [1] World Health Organization et al. Air pollution and child health: prescribing clean air: summary. Technical report, World Health Organization, 2018.
- [2] Sasha Khomenko, Marta Cirach, Evelise Pereira-Barboza, Natalie Mueller, Jose Barrera-Gómez, David Rojas-Rueda, Kees de Hoogh, Gerard Hoek, and Mark Nieuwenhuijsen. Premature mortality due to air pollution in european cities: a health impact assessment. *The Lancet Planetary Health*, 5(3):e121–e134, 2021.
- [3] Randall V Martin, Michael Brauer, Aaron van Donkelaar, Gavin Shaddick, Urvashi Narain, and Sagnik Dey. No one knows which city has the highest concentration of fine particulate matter. *Atmospheric Environment: X*, 3:100040, 2019.
- [4] D Laxen, C Beattie, and F Kirk-Lloyd. Health impacts of air pollution in bristol. *Bristol: Air Quality Consultants*, 2017.
- [5] Tracey Holloway, Daegan Miller, Susan Anenberg, Minghui Diao, Bryan Duncan, Arlene M Fiore, Daven K Henze, Jeremy Hess, Patrick L Kinney, Yang Liu, et al. Satellite monitoring for air quality and health. *Annual Review of Biomedical Data Science*, 4:417–447, 2021.
- [6] Masoud Ghahremanloo, Yannic Lops, Yunsoo Choi, and Bijan Yeganeh. Deep learning estimation of daily ground-level no<sub>2</sub> concentrations from remote sensing data. *Journal of Geophysical Research: Atmospheres*, 126(21), 10 2021. ISSN 2169-897X. doi: 10.1029/2021jd034925. URL <https://www.osti.gov/biblio/1828294>.
- [7] Noel Cressie. The origins of kriging. *Mathematical geology*, 22(3):239–252, 1990.
- [8] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.
- [9] Pratyush Muthukumar, Emmanuel Cocom, Kabir Nagrecha, Dawn Comer, Irene Burga, Jeremy Taub, Chisato Fukuda Calvert, Jeanne Holm, and Mohammad Pourhomayoun. Predicting pm2. 5 atmospheric air pollution using deep learning with meteorological data and ground-based observations and remote-sensing satellite big data. *Air Quality, Atmosphere & Health*, 15(7):1221–1234, 2022.
- [10] Xiaoyan Ma, Jianying Wang, Fangqun Yu, Hailing Jia, and Yanan Hu. Can modis aod be employed to derive pm2. 5 in beijing-tianjin-hebei over china? *Atmospheric Research*, 181: 250–256, 2016.
- [11] Qianqian Yang, Qiangqiang Yuan, Tongwen Li, Huanfeng Shen, and Liangpei Zhang. The relationships between pm2. 5 and meteorological factors in china: seasonal and regional

- variations. *International journal of environmental research and public health*, 14(12):1510, 2017.
- [12] Adven Masih. Application of random forest algorithm to predict the atmospheric concentration of no2. In *2019 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT)*, pages 252–255. IEEE, 2019.
  - [13] Tongwen Li, Yuan Wang, and Qiangqiang Yuan. Remote sensing estimation of regional no2 via space-time neural networks. *Remote Sensing*, 12(16):2514, 2020.
  - [14] MW Gardner and SR Dorling. Neural network modelling and prediction of hourly nox and no2 concentrations in urban air in london. *Atmospheric Environment*, 33(5):709–719, 1999.
  - [15] Masoud Ghahremanloo, Yannic Lops, Yunsoo Choi, and Bijan Yeganeh. Deep learning estimation of daily ground-level no2 concentrations from remote sensing data. *Journal of Geophysical Research: Atmospheres*, 126(21):e2021JD034925, 2021.
  - [16] Bingchun Liu, Xiaogang Yu, Qingshan Wang, Shijie Zhao, and Lei Zhang. A long short-term memory neural network for daily no2 concentration forecasting. *International Journal of Information Technology and Web Engineering (IJITWE)*, 16(4):35–51, 2021.
  - [17] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. *Advances in neural information processing systems*, 31, 2018.
  - [18] Shunchao Yin, Tongwen Li, Xiao Cheng, and Jingan Wu. Remote sensing estimation of surface pm2. 5 concentrations using a deep learning model improved by data augmentation and a particle size constraint. *Atmospheric Environment*, 287:119282, 2022.
  - [19] Nishant Yadav, Meytar Sorek-Hamer, Michael Von Pohle, Ata Akbari Asanjan, Adwait Sahasrabhojanee, Esra Suel, Raphael Arku, Violet Lingenfelter, Michael Brauer, Majid Ezzati, et al. Deep transfer learning on satellite imagery improves air quality estimates in developing nations. *arXiv preprint arXiv:2202.08890*, 2022.
  - [20] Jun Ma, Jack CP Cheng, Changqing Lin, Yi Tan, and Jingcheng Zhang. Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques. *Atmospheric Environment*, 214:116885, 2019.
  - [21] Air quality data continuous, 2022. URL <https://opendata.bristol.gov.uk/explore/dataset/air-quality-data-continuous/information/?disjunctive.location>. [Last accessed 28/10/22].
  - [22] Louise Mittal. London air quality network summary report, 2020. URL [https://londonair.org.uk/london/reports/2020\\_LAQN\\_Report.pdf](https://londonair.org.uk/london/reports/2020_LAQN_Report.pdf). [Last accessed 28/11/22].
  - [23] Joaquín Muñoz-Sabater, Emanuel Dutra, Anna Agustí-Panareda, Clément Albergel, Gabriele Arduini, Gianpaolo Balsamo, Souhail Bousetta, Margarita Choulga, Shaun Harrigan, Hans Hersbach, et al. Era5-land: A state-of-the-art global reanalysis dataset for land applications. *Earth System Science Data*, 13(9):4349–4383, 2021.
  - [24] Ordnance Survey. Os open roads. <https://www.ordnancesurvey.co.uk/business-government/products/open-roads.html>, 2022.
  - [25] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 785–794, 2016.

- [26] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.
- [27] A Jain, I Liu, A Sarda, and P Molino. Food discovery with uber eats: Using graph learning to power recommendations, Dec 2019. URL <https://eng.uber.com/uber-eats-graph-learning/>.
- [28] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 974–983, 2018.
- [29] Lia Chatzidiakou, Anika Krause, Olalekan AM Popoola, Andrea Di Antonio, Mike Kellaway, Yiqun Han, Freya A Squires, Teng Wang, Hanbin Zhang, Qi Wang, et al. Characterising low-cost sensors in highly portable platforms to quantify personal exposure in diverse environments. *Atmospheric measurement techniques*, 12(8):4643–4657, 2019.
- [30] James Thomas and Sam Gunner. Local air – mapping the local environment using e-scooters, 2023. URL <https://www.localair.uk/>.
- [31] CSIRO’s Data61. Stellargraph machine learning library. <https://github.com/stellargraph/stellargraph>, 2018.

Code for all models and plots discussed in this report are available online at <https://github.com/finngueterbock/FYP>