A New Benchmark of Graph Learning for PM2.5 Forecasting under Distribution Shift

Yachuan Liu, Jiaqi Ma, Paramveer Dhillon, Qiaozhu Mei*
{yachuan,jiaqima,dhillonp,qmei}@umich.edu
School of Information, University of Michigan
Ann Arbor, MI, USA

ABSTRACT

We present a new benchmark task for graph-based machine learning, aiming to predict future air quality (PM2.5 concentration) observed by a geographically distributed network of environmental sensors. While prior work has successfully applied Graph Neural Networks (GNNs) on a wide family of spatio-temporal prediction tasks, the new benchmark task introduced here brings a technical challenge that has been less studied in the context of graph-based spatio-temporal learning: distribution shift across a long period of time. An important goal of this paper is to understand the behaviour of spatio-temporal GNNs under distribution shift. To achieve this goal, we conduct a comprehensive comparative study of both graph-based and non-graph-based machine learning models on the proposed benchmark task. To single out the influence of distribution shift on the model performances, we design two data split settings for control experiments. The first setting splits the data naturally by the order of time, while the second setting assigns all the time stamps randomly into training, validation, and test sets, which removes the effect of distribution shift. Our empirical results suggest that GNN models tend to suffer more from distribution shift compared to non-graph-based models, which calls for special attention when deploying spatio-temporal GNNs in practice.

KEYWORDS

graph neural networks, spatial-temporal graph neural networks, PM2.5 forecasting.

ACM Reference Format:
Yachuan Liu, Jiaqi Ma, Paramveer Dhillon, Qiaozhu Mei. 2021. A New Benchmark of Graph Learning for PM2.5 Forecasting under Distribution Shift. In *Proceedings of GLB '21: The Workshop on Graph Learning Benchmarks at the Web Conference 2021. (GLB '21).* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Graph Neural Networks (GNNs) have achieved great success in utilizing neighboring information to generate hidden representations. Because of the ubiquity of graph-structured data, GNNs have been applied to many fields, ranging from social network analysis, protein interaction prediction, and paper citation analysis. At the meantime, various GNNs have been developed to meet different needs [7, 10, 11, 18, 22]. In the past three years, GNNs have been integrated with prediction models designed for spatial data or time series data, such as CNN and LSTM, to provide the spatial and temporal aggregation capability and facilitate various spatial-temporal prediction tasks. Promising results are reported in multiple domains, such as traffic flow prediction and action prediction [14, 23, 25].

In this paper, we present a new air quality forecasting dataset as a novel benchmark task for graph-based spatio-temporal learning. In particular, the air pollutant in our dataset is fine Particulate Matter which has diameters of 2.5 microns or less (PM2.5). Air pollutant data are often collected from environmental sensors or monitoring stations distributed geographically. At each station, the data is represented as a time series. Thus the air pollutant data collected by a monitoring network (e.g., various locations in a city or a state) are naturally spatial-temporal data.

A key feature of this dataset is that the prediction targets have distribution shift across a long period of time, which is a property commonly associated with time-series data. In the case of air quality, the distribution of air pollutant level can be affected by many factors, such as seasonal change, climate change, or social events, etc., and it is difficult to model all such factors through domain knowledge a priori. Therefore, it is important to examine the robustness of temporal prediction models under distribution shift.

In this work, we empirically evaluate the recently developed spatio-temporal GNNs under distribution shift. We deliberately design two data split settings to investigate the influence of distribution shift. In the first setting, we split data into training, validation, and test sets by the order of time, which is a common practice when dealing with time-series data. In the second setting, as a control setup, we randomly split all the time stamps into the three sets regardless of the time order. In this way, we largely remove the distribution shift effect between training and test sets. Our experiment results demonstrate that, in general, the tested spatio-temporal GNNs outperform non-graph-based machine learning methods in the second setting but underperform non-graph-based machine learning methods in the first setting. This phenomenon calls for a special attention to the concern of distribution shift in the deployment of spatio-temporal GNNs.

2 RELATED WORK

In this section, we introduce the related work for spatial-temporal GNNs, PM2.5 prediction, and the existing benchmark datasets for spatio-temporal GNNs.

Spatial-temporal GNNs integrate graph convolution to capture spatial relations with a time series model, such as RNNs or CNN.

GLB '21, April 12–16, 2021, Online

ACM ISBN . . . $
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Graph Convolution Recurrent Network (GCRN) [17] combines ChebNet [3] with LSTM. Difussion Convolution RNN (DCRNN) [13] uses a random walk on the graph to capture spatial diffusion process and embeds it in a GRU model. Spatial-Temporal GCN (STGCN) [25] constructs spatial-temporal blocks by stacking graph convolution and 1D-CNNs. Attention based STGCN (ASTGCN) [6] integrated attetion mechanism both to the spatial and temporal layers. GCRN applied on the image data, while the other models are targeted for traffic prediction.

PM2.5 prediction can be roughly categorized into two approaches: simulation-based methods and data-driven methods. Simulation-based methods, such as CMAQ [5], WRF/CAM [2], and NAQPMS [20], use the knowledge of atmosphere physics and chemical dynamics to get a spatial distribution for air pollutants.

For the data-driven approach, PM2.5 forecasting has two main streams: classical statistical models and machine learning models. Classical statistical methods, represented by ARIMA [21], Kalman Filtering [9], and GTWR [8], have strong assumptions on the data which are often violated by PM2.5 data. Classic Machine learning methods like CNN, RNN and its varation LSTM are adopted particularly for the time series prediction thus are often used on the task of PM2.5 prediction when no spatial dependencies are considered [4, 12]. Recently, there are a few works that apply GNNs to PM2.5 predictions [15, 16, 19, 24, 26]. But their focuses are improving PM2.5 predictions with domain knowledge. This work, instead, focuses on using a PM2.5 benchmark dataset to examine the influence of distribution shift on spatio-temporal GNNs.

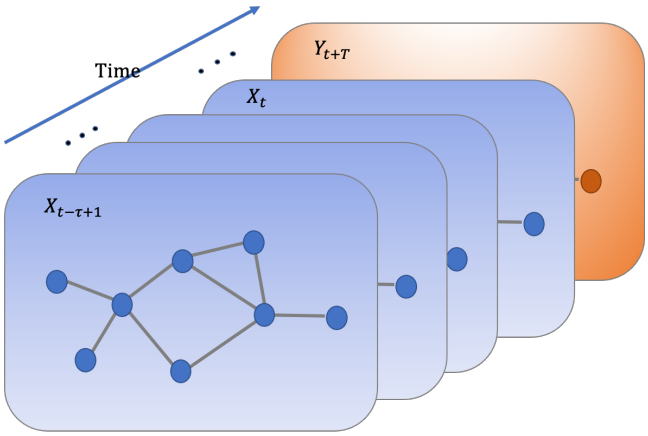

Figure 1: The graph structured data. Each node is a monitoring station. Each slice is a slice in time, X_t stands for the features at time t, and Y_{t+T} is the label. This collection of blue slices is the features for a sample, and the orange slice is the label for a sample.

3 THE PM2.5 FORECASTING BENCHMARK

In this section, we introduce the PM2.5 Forecasting Benchmark, including a formal formulation of PM2.5 forecasting as a prediction problem, the dataset description, and an exploratory analysis.

3.1 Problem Formulation

The goal of PM2.5 forecasting is to use the observed air quality records, meteorological data (e.g. temperature, humidity, and wind levels), and other environmental data from N monitoring stations to forecast the future air PM2.5 concentrations across the area of interest. The monitoring-station network can be presented as an undirected graph $\mathcal{G} = (V, E, A)$. V is the set of nodes (i.e. monitoring stations), and $|V| = N$. E is the set of edges, and A is a weighted adjacency matrix: $A \in \mathbb{R}^{N\times N}$, which is built based on geographical proximity and/or historical correlations between monitoring stations. We assume the monitoring stations record the air quality and other meteorological data at regular time intervals, e.g. every hour. Then at any recorded time stamp t, we have $X_t \in \mathbb{R}^{N\times F}$, where F is the number of features extracted from the recordings, and the PM2.5 concentration $Y_t \in \mathbb{R}^N$. A visualization of the data structure is provided in Figure 1. Note that what we call a *sample* refers to the data (of all stations) associated with a certain time stamp.

At any time point t, given the past τ observations of PM2.5 concentrations and meteorological data at all stations and the monitoring-station network information $G(V, E, A)$, the goal is to forecast the most likely PM2.5 concentration Y_{t+T} at the future time point $t + T$ through a predicting function $h(\cdot)$, i.e.,

$$[\underbrace{X_{t-\tau+1}, ..., X_t}_{\text{past } \tau \text{ observations}}; \mathcal{G}] \xrightarrow{h(\cdot)} \hat{Y}_{t+T}. \tag{1}$$

3.1.1 Graph Construction. Viewing the monitoring stations as nodes of the graph, we want to link two stations if the information from the other station would help the underlying station in the prediction. Considering this, the adjacency matrix A can be constructed through the geographical proximity and historical PM2.5 concentration correlations.

By geographical proximity. The weighted adjacency matrix can be computed based on the distances between monitoring stations. The elements of A could be formed using the Gaussian kernel,

$$A_{ij} = \begin{cases} \exp\left(-\frac{d_{ij}^2}{\sigma^2}\right), i \neq j \text{ and } \exp\left(-\frac{d_{ij}^2}{\sigma^2}\right) \geq \epsilon \\ 0, \text{ otherwise} \end{cases}, \tag{2}$$

where A_{ij} is the weight of edge E_{ij}. σ is the standard deviation of the Gaussian kernel, and ϵ is a threshold that controls graph sparsity.

By historical correlation. The distances between stations sometimes could not correctly reflect the similarity of their PM2.5 concentrations. Exceptions may occur due external geographical features around and in-between two stations, e.g. the terrain. Avoiding this issue, an alternative graph construction approach could utilize the correlation of the historical PM2.5 concentration among stations. A similar graph construction method is used in a bike flow prediction [1]. In this case, the elements of A could be formed as,

$$A_{ij} = \begin{cases} corr_{ij}, i \neq j \text{ and } corr_{ij} \geq \eta \\ 0, \text{ otherwise} \end{cases}, \tag{3}$$

where A_{ij} is the weight of edge E_{ij}. $corr_{ij}$ is the historical correlation of PM2.5 concentration between station i and j, and η is the threshold that controls graph sparsity.

3.2 Dataset Description

The dataset we use is gathered from a major city in Northern China. There are 534 monitoring stations in or around the city. At each station, five features, namely PM2.5, temperature, humidity, wind level, and wind direction are recorded from sensor readings automatically every hour. The geographical location is also associated with each stations. The time span for the dataset is from September 1st, 2018 to December 1st, 2018. We acknowledged that the short time period is a limitation of this dataset. However, the short time span amplifies the distribution shift phenomenon which raises problems on the prediction task applying graph based deep models.

In data prepossessing, we filter out the stations that have more than 30% missing values and there are 415 stations left. Then the missing values are filled by linear interpolation.

For the prediction task, we use observations in the past 24 hours to predict PM2.5 concentration one day ahead for every station, i.e., setting $\tau = 24$ and $T = 24$ in Eq. (1).

Two split settings. We split the dataset into training, validation, and test sets with a proportion 6: 2: 2 under two settings: split-by-time or random-split. The split-by-time setting is what normally used in time series prediction, which segments the dataset with two split points in the time horizon. Under this setting, the training, validation, and test sets cover disjoint time intervals, and the distribution shift problem presents. The random-split setting, on the other hand, shuffles the samples (of different time stamps) and randomly split them into training, validatio, and test sets. Under this setting, the samples can be viewed as independently and identically distributed (i.i.d.). And thus the distribution shift problem among training, validation and test datasets is largely eliminated.

The adjacency matrix is constructed as an average between the adjacency matrices constructed using geographical proximity (Eq. (2)) and historical PM2.5 concentration correlations (Eq. (3)) from all the training samples under the specific training split.

3.3 Exploratory Analysis

We visualize the distribution shift under the split-by-time setting in Figure 2. The upper figure demonstrates the hourly PM2.5 concentration value averaged over all stations, which clearly shows a non-stationary pattern over time. The lower figure further visualizes the histograms of the labels in the training, validation, and test sets, which directly reflects the distribution shift.

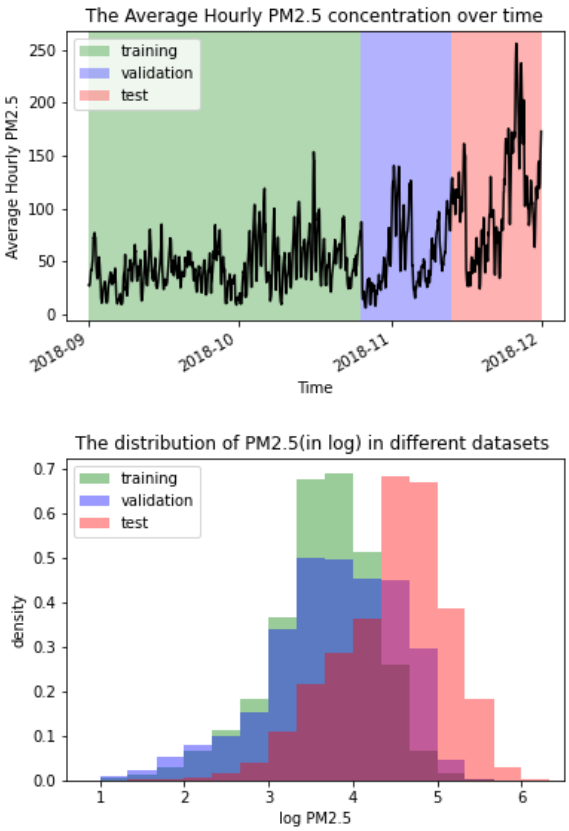

Figure 2: (Under the setting of split by time) The temporal changes which causes distribution shift in training, validation and test datasets. The season changes from fall to winter, and PM2.5 diffusion dynamic could change simultaneously. From the top figure, we see that in validation and test sets, there are several abrupt changes which rarely appear in the training set. From the bottom figure, we see due to the season change, PM2.5 levels are distributed differently in training, validation, and test.

4 BENCHMARK EXPERIMENTS

In this section, we present the benchmark experiments on both graph-based and non-graph-based machine learning models.

4.1 Prediction Models

We compare a wide portfolio of prediction models on this new benchmark dataset:

- **Naive:** Naive model simply uses the current observations as the predictions one day ahead.
- **LR:** Linear regression.
- **ARIMA**: Autoregressive Integrated Moving Average. In this case, since ARIMA can only predict one time step ahead, we use it recursively to get the T steps ahead estimate.
- **MLP:** Multilayer Perceptron of 2 layers with RELU activations. MLP considers each station separately. Therefore the graph structure is not used.
- **GCN**[10]: Graph Convolutional Neural Network with spectral graph kernel with 2 layers. The graph convolution is expected to aggregate spatial information.
- **STGCN**[25]: Spatial-Temporal Graph Convolutional Network, which is originally designed for traffic prediction. The graph convolution could aggregate spatial dependencies and is embedded in a 1-D CNN model.
- **Temporal:** Same model structure as STGCN but without graph convolutions. We feed an adjacency matrix of all zeros

to the STGCN, whcih means we consider all the stations are independent in prediction. Here, Temporal model serves as a comparison for STCGN.

- **ASTGCN**[6]: Attention Based Spatial-Temporal Graph Convolution Network, which combines both spatial attention and temporal attention mechanisms to better capture the spatial-temporal characteristics for data.

For all the non-temporal models, concatenation of the features of the past 24 hours are used as the input.

4.2 Experiment Setup

Under both the split-by-time setting and the random-split setting, each prediction model is trained on the training set, with hyperparameters tuned on the validation set, then tested on the held-out test set.

Since we have a regression problem, the metric we use to evaluate different models performance is RMSE (Root Mean Squared Error), i.e.,

$$RMSE = \sqrt{\frac{1}{nN}\sum_{s=1}^{N}\sum_{t=1}^{n}\left(Y_{st} - \hat{Y}_{st}\right)^2},$$

where $\hat{Y}_{st}$ is the predicted PM2.5 concentration at station s at time t, and Y_{st} is its corresponding actual observation. The lower RMSE, the better model performance.

4.3 Experiment Results

Under the random-split setting. The results under the random-split setting are shown in Table 1. We first observe that STGCN and GCN outperform their non-graph-based counterparts, Temporal and MLP, respectively. In particular, the advantage of GCN over MLP is considerably large. STGCN also gives the best performance among all models. In this case, the use of the graph information seems to be a useful addition to the PM2.5 prediction task.

In addition, we observe that the RMSE scores across the training, validation, and test sets for each model are relatively consistent, indicating the training and validation errors are good estimates of the test error.

Under the split-by-time setting. The results of the split-by-time setting are shown in Table 2. Contrary to the results under the random-split setting, we observe that the RMSE scores across the training, validation, and test sets for each model are drastically different, which is probably due to the effect of distribution shift. Further, a key observation is that the graph-based machine learning models generally underperform their non-graph-based counterparts, indicating the graph-based models suffer more from the distribution shift.

4.4 Discussions

In this section, we provide a (unverified) conjecture further explaining the experiment results as well as some relevant discussions.

One conjecture of why graph-based machine learning models suffer more from the distribution shift is that, unlike other graph structured data such as social networks, the adjacency matrix of monitoring stations are (partly) constructed from the historical data. When there is a distribution shift, the underlying pattern of the graph that was used to be helpful in the predictions also changes, which makes the graph-based machine learning models worse than the non-graph-based models. In other words, the graph built from historical data may not be reliable any more for future uses and could give negative disturbance if there is a shift in the greater environment.

Table 1: RMSE scores of different models under the random-split setting.

Model	Training	Validation	Test
Naive	0.701	0.695	0.700
LR	0.570	0.569	0.568
ARIMA	0.869	0.865	0.864
MLP	0.522	0.522	0.521
GCN	0.418	0.495	0.487
STGCN	0.274	0.308	**0.304**
Temporal	0.284	0.315	0.313
ASTGCN	0.211	0.323	0.344

Table 2: RMSE scores of different models under the split-by-time setting.

Model	Training	Validation	Test
Naive	0.656	0.781	0.702
LR	0.524	0.679	0.623
ARIMA	0.855	0.965	0.968
MLP	0.510	0.659	**0.598**
GCN	0.372	0.709	0.737
STGCN	0.496	0.700	0.812
Temporal	0.488	0.701	0.807
ASTGCN	0.401	0.626	0.680

Recently, the type of graph used for PM2.5 prediction has been expanded from undirected static graph to the directed dynamic graph[19, 26] . This use of learnable dynamic graph could be more adapted to the outer changes and thus retains its positive benefits to the prediction if the pattern changes could be captured exhaustly by the dynamic model. Yet, as mentioned above, these fancier models require expertise knowledge as well as extensive data from various domains. There is still a lack of study on the mitigation from the 'failing loudly' prespective, for example, how to enable the model to give warnings to the outer changes that would make the use of graph hazard rather than helpful.

Finally, a limitation of the dataset used in this study is the relative short time period, which may limit significant improvement on this dataset. However, we believe the finding that graph-based approaches tend to suffer more from distribution shifts is of interest to the community.

5 CONCLUSION

In this paper, we present a new PM2.5 forecasting dataset, featured by distribution shift over time. Using this new benchmark, we evaluate a group of both graph-based and non-graph-based machine learning models under two data split settings, split-by-time and random-split. The first setting presents the distribution

shift challenge while the second setting is designed to eliminate the distribution shift effect. Our experiment results suggest that graph-based machine learning models suffer more from distribution shift. In the future, we plan to gain a better understanding of the underlying mechanism that leads to this phenomenon. We also plan to combine techniques from the distribution shift area with the graph based models to give early warnings once the use of graph is of negative disturbances.

REFERENCES

[1] Di Chai, Leye Wang, and Qiang Yang. 2018. Bike flow prediction with multi-graph convolutional networks. In *Proceedings of the 26th ACM SIGSPATIAL international conference on advances in geographic information systems*. 397–400.

[2] Ming-Tung Chuang, Yang Zhang, and Daiwen Kang. 2011. Application of WRF/Chem-MADRID for real-time air quality forecasting over the Southeastern United States. *Atmospheric environment* 45, 34 (2011), 6241–6250.

[3] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. *arXiv preprint arXiv:1606.09375* (2016).

[4] Junxiang Fan, Qi Li, Junxiong Hou, Xiao Feng, Hamed Karimian, and Shaofu Lin. 2017. A spatiotemporal prediction framework for air pollution based on deep RNN. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 4 (2017), 15.

[5] KM Foley, SJ Roselle, KW Appel, PV Bhave, JE Pleim, TL Otte, R Mathur, G Sarwar, JO Young, RC Gilliam, et al. 2010. Incremental testing of the Community Multiscale Air Quality (CMAQ) modeling system version 4.7. *Geoscientific Model Development* 3, 1 (2010), 205–226.

[6] Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. 2019. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 922–929.

[7] William L Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *NIPS* (2017).

[8] Qingqing He and Bo Huang. 2018. Satellite-based mapping of daily high-resolution ground PM2. 5 in China via space-time regression modeling. *Remote Sensing of Environment* 206 (2018), 72–83.

[9] KI Hoi, KV Yuen, and KM Mok. 2008. Kalman filter based prediction system for wintertime PM10 concentrations in Macau. *Global NEST Journal* 10, 2 (2008), 140–150.

[10] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[11] Johannes Klicpera, Aleksandar Bojchevski, and Stephan Günnemann. 2018. Predict then propagate: Graph neural networks meet personalized pagerank. *arXiv preprint arXiv:1810.05997* (2018).

[12] Xiang Li, Ling Peng, Xiaojing Yao, Shaolong Cui, Yuan Hu, Chengzeng You, and Tianhe Chi. 2017. Long short-term memory neural network for air pollutant concentration predictions: Method development and evaluation. *Environmental pollution* 231 (2017), 997–1004.

[13] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2017. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926* (2017).

[14] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2018. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In *International Conference on Learning Representations*.

[15] Yuxuan Liang, Songyu Ke, Junbo Zhang, Xiuwen Yi, and Yu Zheng. 2018. Geoman: Multi-level attention networks for geo-sensory time series prediction.. In *IJCAI*. 3428–3434.

[16] Yanlin Qi, Qi Li, Hamed Karimian, and Di Liu. 2019. A hybrid model for spatiotemporal forecasting of PM2. 5 based on graph convolutional neural network and long short-term memory. *Science of the Total Environment* 664 (2019), 1–10.

[17] Youngjoo Seo, Michaël Defferrard, Pierre Vandergheynst, and Xavier Bresson. 2018. Structured sequence modeling with graph convolutional recurrent networks. In *International Conference on Neural Information Processing*. Springer, 362–373.

[18] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2018. Graph attention networks. *ICLR* (2018).

[19] Shuo Wang, Yanran Li, Jiang Zhang, Qingye Meng, Lingwei Meng, and Fei Gao. 2020. PM2. 5-GNN: A Domain Knowledge Enhanced Graph Neural Network For PM2. 5 Forecasting. In *Proceedings of the 28th International Conference on Advances in Geographic Information Systems*. 163–166.

[20] Z Wang, T Maeda, M Hayashi, L-F Hsiao, and K-Y Liu. 2001. A nested air quality prediction modeling system for urban and regional scales: Application for high-ozone episode in Taiwan. *Water, Air, and Soil Pollution* 130, 1 (2001), 391–396.

[21] Billy M Williams and Lester A Hoel. 2003. Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results. *Journal of transportation engineering* 129, 6 (2003), 664–672.

[22] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying graph convolutional networks. In *International conference on machine learning*. PMLR, 6861–6871.

[23] Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

[24] Xiuwen Yi, Junbo Zhang, Zhaoyuan Wang, Tianrui Li, and Yu Zheng. 2018. Deep distributed fusion network for air quality prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 965–973.

[25] Bing Yu, Haoteng Yin, and Zhanxing Zhu. 2018. Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. 3634–3640.

[26] Hongye Zhou, Feng Zhang, Zhenhong Du, and Renyi Liu. 2021. Forecasting PM2. 5 using hybrid graph convolution-based model considering dynamic wind-field to offer the benefit of spatial interpretability. *Environmental Pollution* (2021), 116473.

A DATASET

A.1 Overview

The dataset has 534 monitoring stations in or around a major city in China. At each station, five features, namely PM2.5, temperature, humidity, wind level and wind direction is recorded from sensor readings automatically every hour. Wind direction is defined as from which direction the wind is blowing, measured as counter-clockwise to the North. Wind level is describing how intensive the wind is, i.e. it is a generalized measure of wind speed. Humidity is the amount of water vapor in the air. Temperature is the temperature in the air, measured in Celsius. The time span for the dataset is from September 1st, 2018 to December 1st, 2018.
Besides the features, the dataset also provides $'BLOCKID'$ which describes the monitoring stations' unique identification number, as long as the corresponding longitude and latitude.

A.2 Statistic

The detailed statistic for each feature is given in Table 3.

Table 3: Dataset Statistic

	Range	Mean	SD
PM2.5	(3.01, 859.12)	58.21	43.66
Wind Direction	(0, 359)	192.33	107.47
Wind Level	(0, 6)	1.57	0.94
Humidity	(8,100)	58.99	23.55
Temperature	(-4, 36)	15.11	7.25

B HYPER-PARAMETER TUNING

The Hyper-parameters we tuned are listed below:

- **MLP, GCN**:
  number of layers:[2,3,4];
  hidden size for each layer:[16,32,64];
  batch size:[64,128,256,512];
  learning rate:[1e-3,1e-4,1e-5];
  number of epochs: [500,1000,2000].
- **STGCN, Temporal, ASTGCN**:
  hidden sizes:[16,32,64];

batch size:[64,128,256,512];
learning rate:[1e-3,1e-4,1e-5];
number of epochs: [500,1000,2000].