Using Satellite Data on Remote Transportation of Air Pollutants for PM2.5 Prediction in Northern Taiwan

George Kibirige 1,1, Ming-Chuan Yang 2, Chao-Lin Liu 2, and Meng Chang Chen 2

1TIGP Academia Sinica and Department of Computer Science
2Affiliation not available

November 9, 2023

Abstract

We proposed RTP, a composite neural network model that captures knowledge from remote transportation pollution events (RTPEs) to improve the local PM2.5 prediction. To the best of our knowledge, this is the first deep learning work to include knowledge from remote pollutants for PM2.5 prediction. RTP consists of two neural network components: a pre-trained base model and STRI model. The base model captures knowledge from local factors that influence PM2.5 concentrations and STRI captures knowledge from RTPEs by learning spatial-temporal characteristics of Satellite base AOD data and weather features from remote areas. In addition, given the size of the STRI model, to facilitate training and improve results we divide the full STRI model into two components: STRI\_fe, which is used to extract spatial-temporal features from remote areas, and STRI\_sp, which predicts local PM2.5 concentrations using both remote and local features. The prediction results from STRI\_sp show that the prediction error is reduced when remote features are added to the model, demonstrating that the STRI model indeed captures knowledge from RTPEs. To characterize the occurrence of RTPEs in northern Taiwan, we also developed an algorithm to classify PM2.5 concentrations attributable to RTPEs. We use the STRI model for the prediction of two EPA stations located at the northern tip of Taiwan and apply the classification algorithm to the results. This yields improvements in accuracy when remote features are added to the model, which demonstrates the impact of RTPEs at the stations.
Using Satellite Data on Remote Transportation of Air Pollutants for PM_{2.5} Prediction in Northern Taiwan

George William Kibirige, Ming-Chuan Yang, Chao-Lin Liu and Meng Chang Chen

Abstract—Accurate PM_{2.5} prediction is part of the fight against air pollution that helps governments to manage environment policy. Satellite Remote sensing aerosol optical depth (AOD) processed by The Multi-Angle Implementation of Atmospheric Correlation (MAIAC) algorithm allows us to observe the transportation of remote pollutants between regions. Here we proposed a composite neural network model, Remote Transported Pollutants (RTP) model, for such long-range pollutant transportation that predicts more accurate local PM_{2.5} concentrations given such satellite data. The proposed RTP model integrates several deep learning components and learns from the heterogeneous features of various domains. We also detected remote transportation pollution events (RTPEs) at two reference sites from the AOD data. Extensive experiments using real-world data show that the proposed RTP model outperforms the base model that does not account for RTPEs by 17%–20%, 23%–26% and 18%–22% and state-of-the-art models that account for RTPEs by 12%–22%, 12%–14%, and 10%–11% at +4h to +24h, +28h to +48h, and +52h to +72h hours respectively.

Index Terms—Remote transported pollutants(RTP), remote transportation pollution events (RTPEs), deep neural network, PM_{2.5} prediction, composite neural network.

I. INTRODUCTION

With the rapid urban development and industrialization in recent years, the increase in air pollution, leading to health issues such as respiratory and cardiac diseases [1]. Air pollutants consist in part of gaseous and particulate matter (PM). The impacts of PM on human health depend on its size, composition, origin, and solubility. PM_{2.5} has diameter less than 2.5 micrometers (µm), and cannot be filtered by nasal passages, leading to serious respiratory diseases [2]. Many nations have constructed urban stations to monitor the presence of PM_{2.5} in the environment; the resultant datasets can be utilized to better understand and predict PM_{2.5}. In addition, satellites use sensors to gauge the density of pollutants over wide areas and offer updates at regular frequencies. Prediction of future PM_{2.5} levels is a difficult problem [3], as the dispersion of pollutants depends heavily on meteorological features and terrain, in addition to the activities of inhabitants. Prediction is further complicated by factors such as pollutant migration from outside the observed area. Such long-range transport of air pollutants relies on wind and other meteorological effects. Specifically, in this study, we considered pollutants transported from northeastern Asia across the East China Sea to Taiwan.

In this study, due to its coverage area and strong correlation with PM_{2.5} [2], we used aerosol optical depth (AOD) data to better understand the air quality in remote areas. The Moderate Resolution Imaging Spectroradiometer (MODIS) sensor on satellites in the Earth Observing System measures aerosols in the atmosphere. The Multi-Angle Implementation of Atmospheric Correlation (MAIAC) algorithm is used to combine two readings per day from MODIS satellites Aqua and Terra satellites into a single AOD dataset [4]. The MAIAC algorithm has been updated to support global calculations, improving aerosol retrieval, detection of snow and cloud, spatial resolution, and atmospheric correction of the MODIS data. The improved MAIAC product was extensively tested [5] and has been used for air quality and land sensing [6], and has demonstrated improvements in accuracy over MODIS process algorithms such as Dark Target and Deep Blues [7], [8].

The MAIAC products are provided on a 1 kilometer (km) sinusoidal grid, and the gridded data are divided into 1200x1200 square kilometers (km²) standard MODIS tiles [4]. Each tile’s location is represented by horizontal (h) and vertical (v) reading, e.g. h11v05 located at the east coast of United States of America (USA) [4]. In this study, the area in question for extracting remote pollutants consists of four satellite tiles h28v05, h29v05, h28v06 and h29v06 as shown in Fig.1.

In this study we attempted to answer two questions. The first is “Can we identified the occurrence of remote transportation of air pollutants to Taiwan?” We used Tamsui station and Wanli station as reference sites, which located on the northern shore of Taiwan in Fig.1. They are the first to be affected when such long-range air pollutants arrive. We answered this question by detecting PM_{2.5} hikes at either of these sites. The second question is “Can we incorporated knowledge about remote transportation of air pollutants to produce more accurate PM_{2.5} predictions?” To answer this question, we exploit existing neural network models by including an extra neural network model using the data on long-range pollutant transport to predict PM_{2.5} values for Taipei. Answering these questions entails the following challenges in terms of deep learning design and practice.

The first challenge is to prioritize the factors that influence the capture of remote pollutants, as air quality is affected by multiple factors, each with its own spatial and temporal distribution [9]. Second, we incorporated the identified factors into the design of a neural network model to capture the complex interactions between them for better PM_{2.5} prediction [3]. Third, we used model fusion to train the proposed neural network model on large heterogeneous datasets for improved efficiency and prediction results.

Before discussing these challenges, we reviewed useful tools for PM_{2.5} prediction. The influence of remote pollutants on the prediction of future local PM_{2.5} levels involves complex factors such as the spatial-temporal correlation between remote
A deep learning neural network (DNN) can be trained with large meteorological and pollution history datasets to examine this complex correlation between datasets [10], [11]. However, according to [3], for complicated applications in which the problem is not well-defined and is associated with complicated natural environmental factors or social activities, such as long-term (i.e., over the subsequent 48 to 72 hours) prediction of PM$_{2.5}$, deep learning yields poor results. Other than the computationally expensive traditional end-to-end deep learning, one way to obtain reasonably good results is to use a composite neural network [3].

A composite neural network is a collection of pre-trained neural network models that forms a large neural network (NN) to yield greater learning capabilities without the burden of high model training expenses [12]. Yang and Chen predict PM$_{2.5}$ levels using a composite neural network with pre-trained and non-instantiated components using data from sources such as meteorological and pollution history data. In [13], a performance bound is provided for the composition model that yields good performance on PM$_{2.5}$ prediction using real-world data. In this study, we used the composition network model proposed in [3] as a baseline.

We addressed the first challenge by considering the AOD data and weather data of remote areas typically provided in coarse-grained grids. Generally, remote transportation pollution events (RTPEs) are caused by monsoon and frontal surfaces which are synoptic; therefore, we considered wind speed, direction, and related features. AOD data measurements are available from the earth’s surface to the top of the atmosphere, including weather features at different altitudes of atmospheric pressure from 10mb (millibar) to 1000mb to allow deep learning methods to learn temporal and spatial associations.

We tackled the second challenge by extending the pre-trained DNN base model [3] to the remote transported pollutant composite neural network (RTP model), which includes a DNN component to incorporate long-range pollutants for PM$_{2.5}$ prediction [9], [10], [14]–[18]. This new component learns the spatial correlation between remote AOD and local PM$_{2.5}$ and grasps spatial-temporal features from remote areas.

For the third challenge, we break the new large DNN component into two parts: one for feature extraction and one for prediction. This reduces the number of training parameters and thus the computational cost of the training process with virtually equivalent prediction performance.

Apart from the main task, we also improved daily PM$_{2.5}$ prediction by filling a new heterogeneous AOD dataset for Taipei and by changing the topology of the base model [3] to an extended local satellite dataset (ESD) model, which extracts spatial-temporal knowledge from AOD features.

This research work offers the following contributions.

1) To the best of our knowledge, this is the first work to use deep learning to capture extra knowledge from RTPEs from satellite datasets for local PM$_{2.5}$ prediction. The proposed RTP model efficiently captures RTPEs and significantly improves PM$_{2.5}$ prediction in comparison with the base model and state-of-the-art models.

2) This paper addressed challenges using RTPEs as features for local PM$_{2.5}$ prediction. These challenges are addressed in a combined fashion to learn from selected features and models.

3) We developed a classification algorithm to classify RTPEs of two reference sites at different PM$_{2.5}$ levels and increase rates.

4) We applied a composite neural network [3] to develop neural network models incrementally to demonstrate the design rationale and contributions of each component for PM$_{2.5}$ prediction.

In the remainder of this paper, we cover related work in Section II, preliminary knowledge in Section III, the proposed models in Section IV, the classification of RTPEs in Section V, our methodology in IV, datasets and preprocessing in Section VI, and the experiments, results, and discussion in Section VII. Finally, section VIII covers the conclusion.
II. RELATED WORK

This section covers research related to PM$_{2.5}$ prediction using satellite AOD data, RTPEs, and composite neural networks.

Satellite-based AOD measurements have been used to estimate and forecast PM$_{2.5}$ levels due to their high correlation with PM$_{2.5}$ and their large spatial coverage area [2], [19]–[22]. Lee et al. predict daily PM$_{2.5}$ concentrations in the southern and eastern parts of the United States of America (US) at 1km x 1km resolution using AOD data processed by MAIAC, which yields better performance than other algorithms with 10km x 10km resolution [2]. Kloog et al. use MODIS satellite, land usage, and meteorological data for daily prediction of PM$_{2.5}$ [21]. Similarly, Di et al. develop a neural network with convolutional layers to predict PM$_{2.5}$ concentrations over the US. They use convolutional layers to aggregate information in neighboring grid cells to capture spatial and temporal autocorrelations [20]. Chudnovsky et al. study the relationship between PM$_{2.5}$ and AOD, and find a high correlation of determination between them, concluding that AOD data can be used as a proxy for PM$_{2.5}$ ground concentrations [22].

Some use ensemble models for improving PM$_{2.5}$ prediction [23]–[25]. In an ensemble model, a linear combination of the outputs from different individual models is used for PM$_{2.5}$ prediction, yielding results superior to the individual prediction results. Popular ensemble models in a broad sense include AdaBoost (AD) [23], generalized additive models (GAMs) [24], [25], random forests (RFs) [23]–[25], and extreme gradient boosting (XGB) [23]–[25]. Here we compared these ensemble and machine learning models with the proposed RTP model.

Several studies simulate and quantify RTPEs from Asia to Taiwan [26]–[30]. Most use trajectory statistics (TS) and chemical transport modeling (CTM) approaches. TS uses the frequency of backward trajectories in an area to determine whether that area’s pollutants result from remote pollutants. CTM involves a brute-force-based method, which further involves two simulations: one simulation without pollutants from the local area and a normal simulation. The difference between these simulations then determines the pollutant quantities from the remote area. Lin et al. study RTPEs in Taiwan and discover that during winter and spring, 50% to 75% of PM10 in northern Taiwan is due to RTPEs [27]. Chuang et al. study and simulate RTPEs in Taiwan using CTM, showing up to 35% increases in sulfate PM$_{2.5}$ [28]. Similarly, Chen et al. discover that RTPEs are more severe in winter and autumn, increasing the pollutant concentrations by 39% and 41%, respectively [29]. However, approaches which use classical models yield low accuracy for PM$_{2.5}$ prediction due to computational complexity [9] and uncertainty [26]. In this work we used deep neural network models to capture remote pollutants and improve local PM$_{2.5}$ prediction.

One approach for better PM$_{2.5}$ predictions is to use a composite neural network. Yang and Chen predict PM$_{2.5}$ concentrations using a composite neural network base model [3] which consists of six nearest neighbors (NNs) in two groups: pre-trained networks, and non-instantiated networks. The idea behind the base model relies on Yang and Chen's [13] proposal, which focuses on the performance constraints of the two groups w.r.t. PM$_{2.5}$ prediction. The six NNs are connected to form a rooted directed graph, and the objective of which is to produce superior final prediction results. In computational physics, Meng and Karniadakis propose a composite neural network comprised of three NNs, each of which uses a different heterogeneous dataset [12]. We adopted the same approach by extending the base model and forming two composite neural networks.

III. PRELIMINARY KNOWLEDGE

A. Convolutional Neural Network

Convolutional neural networks (CNNs) are popular networks for image classification [20], [31]. A CNN uses a convolutional kernel to scan through the height and width of an image to extract the spatial features of grids within the kernel, as shown in Fig2. It also uses a pooling kernel after convolution to reduce the spatial dimensions of the input image. In this work, we used two-dimension (2-D) convolution to extract spatial features from adjacent grid cells in the satellite image followed by an average pooling layer.

![Fig. 2: CNN with 2x2 convolutional kernel, stride of 2, and 2x2 max pooling kernel on 300x300x1 (height x width x channel) satellite tiles](image)

B. Convolutional LSTM

Convolutional Long Short-Term Memory (ConvLSTM), which combines CNN and long short-term memory (LSTM), is used for spatial-temporal prediction [32]–[34]. LSTM is a recurrent neural network (RNN) structure designed to model sequence or time-dependent behavior. LSTM models long-time dependencies by preventing vanishing gradients during model training [35]. Fig.3 shows the inner structure of a LSTM cell: it consists of an input gate $i_t$, a forget gate $f_t$, and an output gate $o_t$. At time $t$ the cell receives $X_t$ as input, $H_{t-1}$ as the previous state, and $C_{t-1}$ as the previous cell output, and uses these to update the information inside the cell. The cell uses the forget gate to drop unimportant information from the input gate and later uses a series of sigmoid ($σ$) or tanh ($tanh$) activations to update information and produce the current cell state ($H_t$) output through the output gate. ConvLSTM, an extension of LSTM, applies an initial convolutional operation to the input spatial data (typically an image). It uses convolutional operations for input-to-state and state-to-state transitions, producing four-dimensional data that preserves spatial and temporal information. The equations to
formulate ConvLSTM are shown below with a corresponding inner structure in Fig.4. Input $X_t$ and previous cell state $H_{t-1}$ are satellite tiles of the current and previous hour. Below, $\ast$ represents the convolutional operator and $\odot$ denotes the Hadamard product.

\[
\begin{align*}
    i_t &= \sigma(W_{xi} \ast P_q + W_{hi} \ast H_{t-1} + W_{ci} \odot C_{t-1} + b_i) \\
    f_t &= \sigma(W_{xf} \ast P_q + W_{hf} \ast H_{t-1} + W_{cf} \odot C_{t-1} + b_f) \\
    C_t &= \tanh(W_{xc} \ast P_q + W_{hc} \ast H_{t-1} + b_c) \\
    C_t &= f_t \odot C_{t-1} + i_t \odot C_t \\
    o_t &= \sigma(W_{xo} \ast P_q + W_{ho} \ast H_{t-1} + W_{co} \odot C_t + b_o) \\
    H_T &= o_t \odot \tanh(C_t)
\end{align*}
\]

where $W$ represents the weight and $b$ the bias. For instance, for $i_t$ the $\ast$ operator captures spatial features from the input satellite tile $X_t$ and previous cell state $H_{t-1}$ . Input weight $W_{xi}$, previous state weight $W_{hi}$ , and previous cell output weight $W_{ci}$ with input bias $b_i$ are used to process new input information and accumulate them in the cell before forwarding to $f_t$ , which drops unnecessary information from the cell state. Then $C_t$ uses tanh to create a new vector of values after being updated using $\sigma$ . Lastly, $o_t$ uses $\sigma$ on $C_t$ to decide the values of the cell state and $H_t$ uses tanh again to filter other values to produce the final cell state $H_t$ . Here we used ConvLSTM to extract spatial-temporal features from satellite tiles.

IV. PROPOSED MODELS

In this section we describe the proposed composite neural network models ESD and RTP. We explain their components before discussing the main composite models.

A. spatial-temporal remote information neural network Model

The spatial-temporal remote information neural network (STRI), as depicted in Fig.5, predicts the PM$_{2.5}$ concentrations of the 18 Environmental Protection Administration (EPA) stations in Taipei using meteorological and AOD features from remote areas with local meteorological features and PM$_{2.5}$ value. Due to the size of the STRI model with available graphics processing unit (GPU) memory, to reduce computational costs we divide the model into the STRI_fe and STRI_p submodels (or called components). There are two phases of training that the first phase trains the whole STRI model and the second phase fine-tunes the STRI_p model by freezing the STRI_fe model trained in the first phase. In the first training phase, spatial-temporal features from the remote area are extracted by the STRI_fe model, and the STRI_p model learns to predict the PM$_{2.5}$ of Taipei area from the extracted features of the STRI_fe model and its input of local meteorological and PM$_{2.5}$ values. In the second training phase, the STRI_p model is further refined by fixing the STRI_fe model and using the input of local meteorological and PM$_{2.5}$ values. The STRI_p model is relatively small and converges promptly during the second phase of training, providing opportunities for better prediction results.

In the first training phase, the STRI_fe model receives the current input from four satellite tiles, each of which contains 300x300 values downscaled from 1200x1200 during pre-processing, as will be explained in Section VI. The four-dimensional (4d for short) tensor, $[t, c, w, h]$ corresponding to time, channel, width, and height, represents the input shape of each tile. Considering the available memory and computational resources, the model uses average pooling with 3 dimensions(3d) $[c, w, h]$ on each tile to reduce their dimension along the time axis and output $T_q$. The convolutional and pooling layers receive $T_q$ for capturing spatial correlation, and aggregate information between grid cells. The output from the pooling layer on the 4d tensor is denoted by $P_q$:

\[
P_q = L(q(c + b_i \ast v))
\]

where $L$ represents the pooling layer, $c$ is the convolutional feature from the convolutional layer, $b$ is the additional bias, $v$ is a vector with the same size as $c$, and $\varphi$ is an activation function.

\[
c = T_q \ast K
\]

where $T_q$ is the downsampled AOD data, $\ast$ represents the convolutional operation, and $K$ is the convolutional kernel. The STRI_fe model uses a series of ConvLSTM layers with batch normalization, used to speed up the training process [36], in between to capture spatial-temporal information in the grid cells of each tile. The output of ConvLSTM for each tile ($H_{T1}, H_{T2}, H_{T3}, H_{T4}$) are concatenated and then flattened to produce a 1 dimension (1d) tensor $[e_i]$. The flatten layer
unstacks all tensors in 1d, therefore $e$ is results of $(t \cdot c \cdot w \cdot h)$, where $\cdot$ represents multiplication.

On the right hand side of STRI_fe of Fig.5, the ConvLSTM structure with batch normalization is applied to the current remote weather dataset to extract spatial-temporal features, which represent historical weather patterns of wind and other features associated with time and location. Furthermore, the 4d tensor’s output from ConvLSTM, denoted by $H_W$, is flattened and output as 1d tensor $[x_r]$, where $x$ is the product of $t, c, w, h$. After that, the $[e_r]$ and $[x_r]$ tensors are fused together to form 1d tensor $[g_r]$ where $g$ is the sum of $[e]$ and $[x]$. The merged output is denoted by $R_p$, which is the extracted spatial-temporal features of remote pollutants with their corresponding weather features. The STRI_fe model uses RepeatVector to transform $R_p$ to a 2d tensor $[t, g]$ to match output shape $[t, ps]$, where $t$ is the timestamp and $ps$ is the future PM$_{2.5}$ values of 18 stations. Then, $R_p$ which is transferred to the STRI_p model to be concatenated with local PM$_{2.5}$ and meteorology data for PM$_{2.5}$ prediction of the future hours.

In the second training phase, the STRI_p model is further refined with the fixed STRI_fe model to reduce the training time, model complexity, and model parameters for improved prediction results. As shown in Fig.5, the STRI_p model consists of a series of fully connected (FC) layers after the concatenation layers. STRI_p receives a 2d tensor $[t, g]$ as the current extracted spatial-temporal features ($R_p$). It also receives 1d tensors $[pm_r]$ and $[net_r]$ for local sequences of PM$_{2.5}$ ($L_p$) and meteorology data ($L_m$). Future weather forecast data is included in the current $L_m$ to reflect weather fluctuations, because the current weather is not satisfactory for long-term prediction, i.e., beyond 24 hours [9]. Furthermore, the model uses RepeatVector layers on $[pm_r]$ and $[net_r]$ to transform them to 2d tensors $[t, pm]$ and $[t, net]$ to match other input features. Then the input features are merged together, denoted as $HR$, using a concatenation layer to form 2d tensor $[t, feature]$. Finally, FC layers is applied to $HR$ to learn the complex interaction between features extracted from the remote area and local features and make predictions.

$$V_i = \varrho(W_i \odot HR_i + b_i)$$  

where $\varrho$ is the rectified linear activation function (ReLU), $\odot$ denote multiplication and $W_i$ and $b_i$ are the weight and bias of the FC layer of STRI_p. The last FC layer with the linear function performs the prediction and outputs the result, denoted by $y$; $W_c$ and $b_c$ are the weight and bias used by the layer

$$y = W_c \odot V_i + b_e$$

B. Base Model

The composite neural network [3] is comprised of six heterogeneous NNs designed for PM$_{2.5}$ prediction for 18 EPA stations using local influential factors within the Taipei area (30x38 cells). The model uses 21 features from EPA and 26 features from the Center Weather Bureau (CWB) related to air quality and weather status. Among the 1140 cells (i.e., 30x38), there are 18 EPA stations and 33 CWB stations. Before the training phase, Yang and Chen [3] use 4-nearest neighbor (4-NN) clustering to fill grid cells without monitor stations. Fig.6 shows the base model for the next 72hour prediction. For the next 24hour and 48hour predictions, the base model have the same architecture but different details, such as activation functions and weights ($W$).

The base model has ten weight matrices and two activation functions, Linear activation function with bias (linear()) and Scaled Sigmoid activation functions with bias (Scaled_Sigmoid()). The equation below demonstration how
Linear() and Scaled_Sigmoid() work when receiving input $\tilde{h} = (h_1, \ldots, h_n)$.

$$\text{Linear}(\tilde{h}) = \sum_{i=1}^{n} \alpha_i h_i + b \quad (6)$$

$$\text{Scaled_Sigmoid}(\tilde{h}) = \sum_{i=1}^{n} \alpha_i \left( \frac{2}{1 + e^{-2h_i}} - 1 \right) + b \quad (7)$$

where $n$ number of input elements and $\alpha_i$ is the weight of $h_i$ and $b$ is the bias.

The base model combines six heterogeneous models as its components: one LSTM, two FC and three convolutional LSTMs (ConvLSTM). Each component has its own input data and its expected task. For example, ConvLSTMs extract spatial-temporal knowledge of EPA, CWB and weather forecasting datasets, and two FC are expected to automatically distillate the information from EPA and CWB data. The resultant PM$_{2.5}$ prediction performance is the best of all comparisons [3].

**C. Local Satellite Data Model**

The local satellite data (LSD) model is a simpler composite neural network than STRI that LSD only considers local AOD data in the Taipei area. We fill the area with AOD data using satellite tiles h28v06 (tile 1) and h29v06 (tile 2), as discussed in the data preprocessing section. We also fill the Taipei area with PM$_{2.5}$, weather forecast, and meteorological data, all of which are aligned as daily readings.

As shown in Fig.7(a), the model starts with a series of CNNs on the AOD data with the 4d tensor $[t, c, w, h]$ to capture the spatial correlation from neighbors along the temporal axis. Then a pooling layer is applied after CNN to reduce the spatial dimensions and aggregate features between the grids and output $K_o$, a 4d tensor, at the same time the model uses the same series of CNN and pooling layers on the current 4d tensor, which includes meteorological, air quality, and weather forecast data, and outputs 4d tensor ($K_i$). Later, the model flattens $K_o$ and $K_i$ along the temporal axis, yielding two 3d tensors $[t, feature]$ concatenated using an Add layer, producing a 3d tensor. Thirdly the model uses an LSTM to extract temporal related features from this 3d tensor. Finally, the model uses the first FC layer to learn the interaction and correlation between all features in a nonlinear way [9] and then produces the PM$_{2.5}$ prediction using the final FC layer.

**D. Remote Transported Pollutant Model**

RTP is a composite neural network consisting of a pre-trained base model as described in Section IV-B and an STRI_p component, which handles knowledge from Remote Transportation Pollution Events (RTPEs). Fig.7(b) illustrates the RTP structure with both components trained separately, after which they are used as pre-trained components in the RTP model using a series of linear functions linear() for improved overall local PM$_{2.5}$ prediction for the 18 EPA stations.

The RTP model predicts PM$_{2.5}$ concentrations by ensemble input from the two components. The base model yields PM$_{2.5}$ predictions $O = \{o_1, o_2, \ldots, o_n\}_{i=1}^{n}$ and STRI_p yields $X = \{x_1, x_2, \ldots, x_n\}_{i=1}^{n}$ for $n = 18$ monitor stations for next $z$ hour where $z = 4, 8, \ldots, 72$. The RTP model then produces PM$_{2.5}$ predictions for all $n$ stations for the next 1 to $z$ hours using $O$ and $X$. The objective here is to improve local PM$_{2.5}$ prediction by accounting for RTPEs.

**E. Extended Local Satellite Model**

As shown in Fig.7(c), the ESD model with series of Linear() is composed of the base model and the LSD model, which handles heterogeneous satellite AOD data. AOD sensory data is composed of columnar pollutant measurements as opposed to ground measurements, and due to the difference of granularity, AOD is interpolated for the finer granularity in the Taipei region. The difference between the RTP and ESD models is that ESD has the LSD component with local AOD knowledge to improve daily PM$_{2.5}$ prediction, while RTP utilizes STRI_p, which learns the remote AOD knowledge.
ESD produces predictions using these two components. Given the PM$_{2.5}$ prediction output $P = \{p_1, p_2, \ldots, p_n\}_{d=1}^f$ of next $f$ days for cells with $n$ monitor station from the base model, where $f = \{1, 2, 3\}$, as well as the PM$_{2.5}$ prediction output $Y = \{y_1, y_2, \ldots, y_n\}_{d=1}^f$ of next $f$ days for the same $n$ cells from the LSD model, then the ESD predicts the PM$_{2.5}$ concentrations of the next $f$ days of all $n$ cells.

V. CLASSIFICATION OF REMOTE POLLUTANTS

In this section, we answered the first question by classifying PM$_{2.5}$ levels at the Wanli and Tamsui stations in the Taipei region that are affected by RTPEs. Again, as shown in Fig.1, due to their locations in Taiwan, these stations are the first to show evidence of RTPEs. In addition, since both Wanli and Tamsui are by the seashore, their background PM$_{2.5}$ values are stable and relatively low. The occurrence of RTPE will cause the rise of their PM$_{2.5}$ value. Therefore, we start by producing PM$_{2.5}$ predictions using the STRI_fe and SRTI_p models without considering the local PM$_{2.5}$ influence factor. Then we use a classification algorithm for different PM$_{2.5}$ concentrations in the prediction results. In our prediction experiment, we considered only from November to May because they are months when RTPEs have the greatest impacts on northern Taiwan [26]. The STRI_fe model is discussed in the previous section; here we cover the classification algorithm.

The classification algorithm classifies the PM$_{2.5}$ concentrations for the next 24, 48, and 72 hours that are affected by RTPEs. Such remote pollutants are understood to flow across the eastern China Sea to Taiwan; however, due to variations in wind direction, not all pollutants reach Taiwan. Thus, we seek to ascertain the amount of pollutants reaching Taiwan. In the first condition, we assume that the arrival of such remote pollutants raises PM$_{2.5}$ concentrations beyond a certain threshold at these stations. In the second condition, we focused on PM$_{2.5}$ increases over two consecutive hours because we assume that remote pollutants increase rather than decrease the concentration of local pollutants. That is, our second assumption is that remote pollutants yield positive changes between two consecutive predictions. For example, if remote pollutants arrive in the current hour ($t$), then the difference between the current PM$_{2.5}$ and that of the previous hour ($t$-1) must be positive.

A. Classifying remote transportation pollution events

For the two conditions, we created three thresholds each. That for the first condition is the EPA threshold (Epa_tshd). Chuang et al. show that RTPEs in northern Taiwan account for PM$_{2.5}$ concentrations ranging from 31 to 39 [26]. These quantities were obtained by studying the contribution of RTPEs on PM$_{2.5}$ in northern Taiwan during the monsoon winter seasons from 2005 to 2015. We used this range to select thresholds of 30, 33, and 36 for the first condition. Predictions exceed any of these thresholds are said to satisfy the first condition.

For the second condition, we considered the differential threshold (Diff_tshd). We convert the true and predicted PM$_{2.5}$ concentrations to first-order difference vectors, after which we select differential thresholds 0.5, 1.0, and 1.5. For the second condition the difference between the current hour’s predicted value and that of the previous hour must be greater than or equal to these Diff_tshd. Therefore, an RTPE is said to occur if the peak value simultaneously satisfies both conditions.

Fig.8 is a figure of an example with the ground-truth (GT) and predicted results of the next one hour and next four hours for the Wanli station. The Epa_tshd and Diff_tshd used of the example are 30 and 0.5, respectively. The green dashed line indicates Epa_tshd, and the colored dots represent the peaks from different predicted hours that exceed the two thresholds. The 26 red dots represent the total number of RTPEs predicted in 1 hour, compared to the 69 ground-truth events. The accuracy of RTPE detection is thus 37.7%.

\[ F1_{score} = \frac{(2 \times TP)}{((2 \times TP) + FN + FP)} \]  

For the formulas for A, Pr, and R, see the deep learning textbook [37]; further classification details are provided in the Appendix.

VI. DATASETS AND DATA PREPROCESSING

Of the two dataset groups, that used to evaluate the ESD model contains daily data and that for the RTP model contains hourly data.

A. Dataset for ESD Model

The dataset for the ESD model includes daily satellite AOD data, air quality data, and meteorological data. The model is trained on two years (2014 and 2015) of data and tested on one year (2016) of data.
1) Satellite MAIAC AOD dataset (daily): This is satellite AOD data at a 1x1-km resolution created using the MAIAC algorithm and downloaded from the National Aeronautics and Space Administration (NASA) website [38]. Taipei, the area in question, with an area of 30x38km², is located between tiles 1 and 2 (h28v06 and h29v06), each of which covers a 1200x1200km² area. Fig.1 shows these two tiles, which we used to fill the AOD data of Taipei area by matching their latitude and longitude coordinates. After matching these coordinates, we pre-processed and filled grid cells with missing or anomalous AOD data with the means of AOD data of its 3x3 neighbouring cells.

2) Air quality dataset (daily): We obtained hourly air quality data from the Taiwan EPA website [39] consisting of PM_{10}, PM_{2.5}, Carbon monoxide (CO), Nitrogen Oxides (NOx), Ozone (O₃), and Sulfur dioxide (SO₂). In the study area there are a total of 1140 grid cells (30x38), eighteen of which have EPA monitor stations; thus we use the four nearest neighbours (4-NN) method to fill grid cells with empty values. We then convert all data to daily readings by taking 24-hour averages.

3) Meteorological Data (daily): We obtained meteorological data from the Center Weather Bureau (CWB) website [40] which is provided at six-hour intervals. Each reading includes wind speed and direction, rainfall, pressure, temperature, and humidity. The data covers only 33 grid cells in the Taipei area. Again, we use 4-NN to fill those cells without monitor stations and then average these into daily readings.

B. RTP model dataset

1) Satellite MAIAC AOD dataset (per day): As shown in Fig.1, the remote area is covered by satellite tiles h28v05, h29v05, h28v06, and h29v06 with the MAIAC AOD sensory data and each tile covers a 1200x1200-km² area, for a total of 1200x1200 grid cells per tile. The satellite AOD were obtained from the NASA website [38].

First, we calculate the daily means of quality AOD values to fill missing and poor-quality AOD. If there is no quality AOD data for an entire day, we use the mean values of the previous day. We assume AOD values of grid cells over a day are the same for the whole day; thus we repeat the same value 24 times to match 24-hour readings per day.

Secondly, to make best use of the available GPU memory and to reduce computational costs, we downscale each tile separately to reduce the spatial dimensions. Sønderby et al. downscale satellite images for precipitation from a spatial dimension of 1024x1024km² to 512x512km² using mean pooling [33]. Here we used the same approach; however, to maintain the distribution of values in each tile, we use maximum pooling instead of mean pooling. Maximum pooling is applied twice to each tile image to downscale it from 1200x1200km² to 300x300km².

2) Remote Meteorological Data (hourly): We used the National Center for Environmental Prediction (NCEP) final (FNL) global analysis data [41]. These are collected all over the world and provided by the Global Data Assimilation System, and are provided over 28x28-km² grids every six hours. We converted these to one-hour intervals via linear interpolation to match the other data. The data include the following meteorological features: temperature, pressure, vertical velocity (VVEL), absolute vorticity (ABSV), lifted index (LFTX), wind speed, and wind direction. The wind speed (denoted as \( ws \)) and direction (denoted as \( \theta \)) are represented as \( u \) and \( v \) components, i.e., \( ws \times \cos(\theta) \) and \( ws \times \sin(\theta) \). The \( u \) component is the horizontal speed toward the east (known as Zonal Velocity) and \( v \) component is the horizontal wind speed toward the north (known as Meridional Velocity). The wind speed and direction, temperature, VVEL and ABSV were considered at pressure levels from 10mb to 1000mb.

3) Air quality dataset (hourly): We obtained PM_{2.5} features for the 18 stations in Taipei from the EPA website [39]. The data were provided at one-hour intervals.

4) Local Meteorological Data (hourly): We obtained meteorological data for the 18 stations in Taipei from the CWB website [40]. The data were provided at six-hour intervals, and we transformed them to hourly intervals via linear interpolation. The data consist of features for rainfall, pressure, temperature, humidity, wind speed, and wind direction.

VII. EXPERIMENTAL RESULTS AND DISCUSSION

A. Experimental settings and model parameters

We used data from 2014 to 2016 to evaluate the proposed neural network models. Data of two years (2014–2015) were used for training and one year (2016) for testing. The data for the ESD evaluation were prepared at a daily granularity, whereas for the RTP model the data were at hourly granularity.

The two models were trained on an NVIDIA GPU with 11 gigabytes (GB) and implemented on Keras with Tensor Flow backend environment. In the proposed models, we used the rectified linear activation function (Relu) for all layers except for the prediction layer. Learning rates of 0.0015 and 0.0001 were used for ESD and RTP model, respectively; 0.00001 was used for the STRI_fe and STRI_p models, and Adam optimization was used for all training. In addition, to reduce overfitting, we applied dropout layers, weights and bias kernel regularizations to several layers.

All models were evaluated using root mean square error (RMSE), which shows the difference between the predicted and true values

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

where \( n \) is the number of stations and \( y \) and \( \hat{y} \) are true values and predicted value at timestamp. In this work, we consider the mean RMSE of all monitor stations.

B. Performance of ESD model

In Table I we compare the PM_{2.5} prediction results of the ESD model with that of its components, base model and the LSD model, where \( \Delta \% \) is the relative improvement in RMSE over the base model. We also include the performance of RTP model which performed in daily like other models in table. The RTP model that use RTPes from 2tile and 4tile are represented by RTP_2tile and RTP_4tile respectively.
We begin by comparing LSD, ESD and the base model and then we compare them with RTP model. For the base model, a large model with six components, converting the hourly data to daily data reduces the amount of training data, which impacts the base model training, especially for long-term prediction. The base model outperforms the LSD model at the next 1 day (+1day in short) because it uses more features than LSD. For +2day and +3day, the LSD component outperforms the base model due to the application of AOD data in the models. For the proposed ESD model, which consists of the base model with the addition of the pre-trained LSD component, the corresponding heterogeneous AOD data improves RMSE prediction by 12.68%, 11.45%, and 6.65% for +1day, +2day, and +3day, respectively, in comparison to the base model. The topological changes of ESD with the addition of new local AOD knowledge decrease the prediction error between the true and predicted values.

For comparison of RTP model with base model, LSD model and ESD model. The RTP 2tile and RTP 4tile outperform all those three models for +1day, +2day and +3day. The results show that the RTP model captures RTPEs from remote AOD and weather data and it helps improve the prediction performance for all days. With reference to the base model, the RTP 4tile provides the greatest improvement prediction performance in RMSE by 25.77% for +1day and 28.96% and 21.17% for +2day and +3day. In other words, the RTP 4tile outperforms RTP 2tile in all three days that the result demonstrates the enlarged remote area will help improve the local prediction of PM2.5. This matches with our idea of enlarging the remote area from 2tile to 4tiles with the objective of capturing more RTPEs.

| Target | Model | RMSE  | Δ(%)   |
|--------|-------|-------|--------|
| +1day  | base  | 7.299 |        |
|        | LSD   | 8.170 | -11.93 |
|        | ESD   | 6.373 | 12.68  |
|        | RTP_2tile | 5.781 | 18.58  |
|        | RTP_4tile | 5.418 | 25.77  |
| +2day  | base  | 9.802 |        |
|        | LSD   | 8.801 | 10.21  |
|        | ESD   | 8.679 | 11.45  |
|        | RTP_2tile | 8.342 | 14.89  |
|        | RTP_4tile | 6.963 | 28.96  |
| +3day  | base  | 9.898 |        |
|        | LSD   | 9.389 | 5.14   |
|        | ESD   | 9.240 | 6.65   |
|        | RTP_2tile | 8.523 | 13.89  |
|        | RTP_4tile | 7.803 | 21.17  |

C. Prediction of RTPEs

To answer the first question that we raised in the Introduction, i.e., to predict RTPEs, we predicted the local PM2.5 for the two stations first using only the local PM2.5 and weather as input to the STRI_p model with the extracted spatial-temporal features from remote areas. We predicted the RTPEs by applying the thresholds Diff_tshd and Epa_tshd to the PM2.5 predictions. In order to observe the general performance, we used combinations of various thresholds that Diff_tshd=30, 33, and 36 and Epa_tshd=0.5, 1.0, 1.5. Tables II to IV show the classification results in terms of accuracy (A), precision (Pr), recall (R), and F1 score (F1). The first column indicates the data used, for instance, “P” represents the local PM2.5 values, “EP” represents the remote spatial-temporal features from four tiles, and “W” represents the local weather features. The results are shown in Tables II, III and IV.

D. Performance of RTP Model

In this section we answer the second question about improving local PM2.5 predictions using knowledge about RTPEs. We discuss the results of different training approaches for the RTP components, knowledge captured from RTPEs, and the results of RTP models in comparison to other models.

We first evaluated the effect of training strategies on prediction performance by comparing the results from a full STRI model with those using the STRI_fe and STRI_p components, as described in Section IV. Fig.9 shows the comparison results of STRI and STRI_p from Next 4hour (+4h in short) to Next 72hour (+72h in short). STRI_p yields better prediction results than the full STRI model. As training a full STRI model on a single GPU can be a challenge, we use STRI_fe for feature extraction and STRI_p for prediction. Furthermore, as STRI_p consists of a small number of layers, it converges quickly during training, leaving more room for model fine-tuning. The improved performance of STRI_p validates our idea of breaking the full STRI model into two components.

We also evaluated the effects of the extracted remote pollutants and local features on the STRI p component to show whether the proposed STRI model is able to capture knowledge from remote areas. We conducted experiments using one feature and incrementally added features while observing the results in terms of root-mean-square error (RMSE). Fig.10 shows the results of various features, including spatial-
temporal features from two and four tiles (t12 and t1234) as well as the local PM2.5 (P) and weather (W) features from 18 stations. The weather features include the current and forecasted weather. Therefore, the model input sequence is P, t12 (tiles h28v06 and h29v06), the remaining two tiles t34 (tiles h28v05 and h29v05), and then W. Thus, P+t1234 indicates that P and t1234 are used as the model input.

In Fig.10, we observed a significant gap between the performance when using P and that when using data on remote pollutants from tiles t12 (P+t12) and t1234 (P+t1234). This shows that the model captures RTPEs and that these events improve the local prediction performance by reducing the prediction error for all hours. We also observed the impact from the expanding the range from tiles t12 to t1234, mainly for +28h and longer. This impact is not present between +4h and +24h, possibly as events from the extra two tiles (t34) require additional time to make an impact, due to their distance from the study area. This fits with our goal of expanding the range to

### TABLE II: CLASSIFICATION RESULTS WITH $tshd = 0.5$

| $tshd$ | Wanli Station | +24h | +48h | +72h |
|--------|---------------|------|------|------|
| $tshd$ | A | Pr | R | F1 | A | Pr | R | F1 | A | Pr | R | F1 |
| P | 0.44 | 0.20 | 0.32 | 0.25 | 0.19 | 0.10 | 0.17 | 0.12 | 0.08 | 0.06 | 0.08 | 0.07 |
| P+EP | 0.71 | 0.20 | 0.43 | 0.28 | 0.26 | 0.12 | 0.22 | 0.16 | 0.06 | 0.05 | 0.06 | 0.06 |
| P+Ep+W | 0.72 | 0.23 | 0.44 | 0.30 | 0.33 | 0.14 | 0.26 | 0.18 | 0.21 | 0.12 | 0.18 | 0.14 |

| $tshd$ | Kaiji Station | +24h | +48h | +72h |
|--------|---------------|------|------|------|
| $tshd$ | A | Pr | R | F1 | A | Pr | R | F1 | A | Pr | R | F1 |
| P | 0.37 | 0.22 | 0.29 | 0.25 | 0.14 | 0.10 | 0.14 | 0.12 | 0.02 | 0.03 | 0.05 | 0.03 |
| P+EP | 0.61 | 0.22 | 0.41 | 0.29 | 0.10 | 0.08 | 0.10 | 0.09 | 0.02 | 0.02 | 0.02 | 0.02 |
| P+Ep+W | 0.62 | 0.24 | 0.41 | 0.30 | 0.16 | 0.12 | 0.15 | 0.14 | 0.08 | 0.08 | 0.08 | 0.08 |

### TABLE III: CLASSIFICATION RESULTS WITH $tshd = 1.0$

| $tshd$ | Wanli Station | +24h | +48h | +72h |
|--------|---------------|------|------|------|
| $tshd$ | A | Pr | R | F1 | A | Pr | R | F1 | A | Pr | R | F1 |
| P | 0.30 | 0.20 | 0.25 | 0.22 | 0.11 | 0.10 | 0.11 | 0.10 | 0.02 | 0.03 | 0.02 | 0.02 |
| P+EP | 0.45 | 0.22 | 0.34 | 0.27 | 0.02 | 0.06 | 0.02 | 0.03 | 0.01 | 0.03 | 0.01 | 0.02 |
| P+Ep+W | 0.46 | 0.24 | 0.34 | 0.28 | 0.06 | 0.08 | 0.06 | 0.07 | 0.03 | 0.04 | 0.03 | 0.04 |

| $tshd$ | Kaiji Station | +24h | +48h | +72h |
|--------|---------------|------|------|------|
| $tshd$ | A | Pr | R | F1 | A | Pr | R | F1 | A | Pr | R | F1 |
| P | 0.50 | 0.25 | 0.35 | 0.29 | 0.32 | 0.18 | 0.26 | 0.21 | 0.19 | 0.16 | 0.17 | 0.16 |
| P+EP | 0.66 | 0.28 | 0.42 | 0.34 | 0.32 | 0.20 | 0.26 | 0.22 | 0.19 | 0.16 | 0.17 | 0.16 |
| P+Ep+W | 0.63 | 0.29 | 0.41 | 0.34 | 0.41 | 0.22 | 0.31 | 0.25 | 0.24 | 0.17 | 0.20 | 0.19 |

| $tshd$ | Tamsui Station | +24h | +48h | +72h |
|--------|---------------|------|------|------|
| $tshd$ | A | Pr | R | F1 | A | Pr | R | F1 | A | Pr | R | F1 |
| P | 0.38 | 0.23 | 0.29 | 0.26 | 0.20 | 0.13 | 0.17 | 0.15 | 0.09 | 0.10 | 0.08 | 0.09 |
| P+EP | 0.54 | 0.27 | 0.36 | 0.31 | 0.15 | 0.15 | 0.14 | 0.14 | 0.11 | 0.14 | 0.10 | 0.12 |
| P+Ep+W | 0.51 | 0.29 | 0.35 | 0.32 | 0.18 | 0.17 | 0.16 | 0.17 | 0.13 | 0.17 | 0.12 | 0.14 |

| $tshd$ | Kaiji Station | +24h | +48h | +72h |
|--------|---------------|------|------|------|
| $tshd$ | A | Pr | R | F1 | A | Pr | R | F1 | A | Pr | R | F1 |
| P | 0.29 | 0.22 | 0.24 | 0.23 | 0.08 | 0.09 | 0.08 | 0.09 | 0.02 | 0.05 | 0.02 | 0.03 |
| P+EP | 0.38 | 0.27 | 0.30 | 0.28 | 0.03 | 0.05 | 0.04 | 0.04 | 0.05 | 0.09 | 0.05 | 0.06 |
| P+Ep+W | 0.37 | 0.30 | 0.29 | 0.29 | 0.07 | 0.15 | 0.08 | 0.10 | 0.06 | 0.13 | 0.06 | 0.08 |
### Table IV: Classification Results with Diff_thld = 1.5

|            | Wanli Station |                                |                                |
|------------|---------------|--------------------------------|--------------------------------|
|            | Epa_thld      | +24h                          | +48h                          |
|            |               | A Pr R F1                      | A Pr R F1                      |
| P          | 0.44          | 0.17                          | 0.32                          | 0.22                          |
|            |               | 0.17                          | 0.07                          | 0.16                          | 0.10                          |
|            | P+EP          | 0.68                          | 0.17                          | 0.43                          | 0.24                          |
|            |               | 0.20                          | 0.09                          | 0.18                          | 0.12                          |
|            | P+EP+W        | 0.66                          | 0.18                          | 0.42                          | 0.26                          |
|            |               | 0.33                          | 0.12                          | 0.26                          | 0.16                          |
| Epa_thld   |               | 30                            |                                |                                |
| P          | 0.39          | 0.18                          | 0.30                          | 0.23                          |
|            |               | 0.10                          | 0.06                          | 0.10                          | 0.08                          |
|            | P+EP          | 0.63                          | 0.20                          | 0.41                          | 0.26                          |
|            |               | 0.08                          | 0.07                          | 0.08                          | 0.07                          |
|            | P+EP+W        | 0.59                          | 0.20                          | 0.39                          | 0.27                          |
|            |               | 0.16                          | 0.10                          | 0.15                          | 0.12                          |
|            |               |                                |                                |                                |
| Epa_thld   | 33            |                                |                                |                                |
| P          | 0.29          | 0.15                          | 0.25                          | 0.19                          |
|            |               | 0.09                          | 0.07                          | 0.09                          | 0.08                          |
|            | P+EP          | 0.43                          | 0.19                          | 0.33                          | 0.24                          |
|            |               | 0.02                          | 0.05                          | 0.02                          | 0.03                          |
|            | P+EP+W        | 0.44                          | 0.21                          | 0.33                          | 0.26                          |
|            |               | 0.06                          | 0.08                          | 0.07                          | 0.07                          |

|            | Tamsui Station |                                |                                |
|------------|---------------|--------------------------------|--------------------------------|
|            | Epa_thld      | +24h                          | +48h                          |
|            |               | A Pr R F1                      | A Pr R F1                      |
| P          | 0.46          | 0.20                          | 0.34                          | 0.25                          |
|            |               | 0.30                          | 0.14                          | 0.25                          | 0.18                          |
|            | P+EP          | 0.63                          | 0.24                          | 0.41                          | 0.30                          |
|            |               | 0.29                          | 0.16                          | 0.24                          | 0.19                          |
|            | P+EP+W        | 0.61                          | 0.25                          | 0.40                          | 0.31                          |
|            |               | 0.38                          | 0.18                          | 0.30                          | 0.22                          |
| Epa_thld   | 36            |                                |                                |                                |
| P          | 0.37          | 0.20                          | 0.28                          | 0.23                          |
|            |               | 0.18                          | 0.11                          | 0.16                          | 0.13                          |
|            | P+EP          | 0.49                          | 0.22                          | 0.34                          | 0.27                          |
|            |               | 0.10                          | 0.11                          | 0.10                          | 0.10                          |
|            | P+EP+W        | 0.49                          | 0.25                          | 0.34                          | 0.29                          |
|            |               | 0.16                          | 0.14                          | 0.15                          | 0.14                          |
| Epa_thld   | 36            |                                |                                |                                |
| P          | 0.27          | 0.18                          | 0.23                          | 0.20                          |
|            |               | 0.05                          | 0.06                          | 0.06                          | 0.06                          |
|            | P+EP          | 0.32                          | 0.22                          | 0.27                          | 0.24                          |
|            |               | 0.03                          | 0.04                          | 0.03                          | 0.04                          |
|            | P+EP+W        | 0.32                          | 0.24                          | 0.27                          | 0.25                          |
|            |               | 0.07                          | 0.12                          | 0.07                          | 0.09                          |

---

Four tiles to improve prediction by capturing more RTPEs. The addition of W yields superior performance, especially beyond +28h. We attribute this gap to long-term rather than short-term (+4h to +24h) weather fluctuations; that is, in the long term, the weather forecast is used to reflect weather fluctuations [9], which explains the significant gap in that period. Overall, the results show that the STRI_fe component captures knowledge from RTPEs by learning spatial-temporal behavior from AOD data with their corresponding weather features from remote areas before transferring to the STRI_p component, which then learns the complex interaction between all input features, yielding improved local PM_{2.5} prediction for the 18 stations.

Third, we evaluated the performance of the RTP model in comparison with the base model [3] and other state-of-the-art ensemble models with the same settings: linear regression (LR) [16], AdaBoost (AB) [23], bagging regression (BG), random forest (RF), extreme gradient boosting (XGB) [23–25], and a generalized additive model (GAM) [24], [25]. We also show RTP performance when we use RTPEs from 2tile(RTP_{2tile}) and 4tile(RTP_{4tile}) with the objective of showing the impact of remote pollutants towards local prediction of PM_{2.5}.

Fig. 11 shows the relative prediction improvements in RMSE of both RTP models and the other state-of-the-art models w.r.t. base model from +4h to +72h; the greater the improvement, the better the model does in comparison to the base model. The figure shows that RTP_{4tile} yields the greatest improvements: from 17%–30%, 23%–26%, and 18%–22% for +4h to +24h, +28h to +48h, and +52h to +72h, respectively. Similarly, the RTP_{2tile} provides greater improvement: from 13%–24%, 17%–23%, and 13%–17% for +4h to +24h, +28h to +48h, and +52h to +72h. XGB and GAM, in turn, improve on the base model by 6%–8%, 10%–12%, and 8%–11% for +4h to +72h.
+24h, +28h to +48h, and +52h to +72h, respectively, with scores that are similar to those for the LR model. AB is outperformed by the base model for most hours; RF is also, but to a lesser extent. The RTP_2tile and RTP_4tile outperform other models due to its composite neural network design [13], which involves high flexibility with learning capability to model nonlinear association between input features. On the other hand, for XGB, GAM and LR yield good performance improvements but are outperformed by both RTP models because of their limited flexibility in their model structure as a result of insufficient learning capability from input features. The performance of RTP_4tile over RTP_2tile continues to demonstrate the importance of the enlarged remote area with the aim of capturing more RTPEs from the remote area.

**Fig. 11:** Relative RMSE Improvement(%) of all models with reference to base model

### VIII. Conclusion

We proposed RTP, a composite neural network model that captures knowledge from remote transportation pollution events (RTPEs) to improve the local PM$_{2.5}$ prediction. To the best of our knowledge, this is the first deep learning work to include knowledge from remote pollutants for PM$_{2.5}$ prediction. RTP consists of two neural network components: a pre-trained base model and STRI model. The base model captures knowledge from local factors that influence PM$_{2.5}$ concentrations and STRI captures knowledge from RTPEs by learning spatial-temporal characteristics of AOD data and weather features from remote areas. In addition, given the size of the STRI model, to facilitate training and improve results we divide the full STRI model into two components: STRI$_{2 tile}$, which is used to extract spatial-temporal features from remote areas, and STRI$_{4 tile}$, which predicts local PM$_{2.5}$ concentrations using both remote and local features. The prediction results from STRI$_{4 tile}$ show that the prediction error is reduced when remote features are added to the model, demonstrating that the STRI model indeed captures knowledge from RTPEs.

To characterize the occurrence of RTPEs in northern Taiwan, we also developed an algorithm to classify PM$_{2.5}$ concentrations attributable to RTPEs. We use the STRI model for the prediction of two EPA stations located at the northern tip of Taiwan and apply the classification algorithm to the results. This yields improvements in accuracy when remote features are added to the model, which demonstrates the impact of RTPEs at the stations.

We perform local PM$_{2.5}$ prediction using the RTP model for all stations from +4h to +72h. The RMSE results show that the RTP model outperforms the base model and other state-of-the-art ensemble models by 12%–30%, 12%–18%, and 10%–14% for +4h to +24h, +28h to +48h, and +52h to +72h, respectively, due to RTP’s composite neural network [13]. The ESD model, although it just considers local AOD data, still improves PM$_{2.5}$ prediction, as evidenced by lower RMSE scores than the those for the base model by 12.68% for +1day and 11.45% and 6.65% for +2day and +3day.

Future work will focus on expanding the remote area, using data that is updated at a higher frequency compared to the AOD data, considering other possible features and re-analyzing weather features.

### References

[1] J. S. Pandey, R. Kumar, and S. Devotta, “Health risks of NO2, SPM and SO2 in Delhi (India),” *Atmospheric Environment*, vol. 39, no. 36, pp. 6868–6874, 2005.

[2] M. Lee, I. Kloo, A. Chudnovsky, A. Lyapustin, Y. Wang, S. Melly, B. Coull, P. Koutrakis, and J. Schwartz, “Spatiotemporal prediction of fine particulate matter using high-resolution satellite images in the Southeastern US 2003–2011,” *Journal of exposure science & environmental epidemiology*, vol. 26, no. 4, pp. 377–384, 2016.

[3] M.-C. Yang and M. C. Chen, “PM$_{2.5}$ Forecasting Using Pre-trained Components,” in *2018 IEEE International Conference on Big Data (Big Data)*. IEEE, 2018, pp. 4488–4491.

[4] A. Lyapustin, Y. Wang, S. Korkin, and D. Huang, “MODIS Collection 6 MAIAC algorithm.” *Atmospheric Measurement Techniques*, vol. 11, no. 10, 2018.

[5] T. Hilker, A. I. Lyapustin, C. J. Tucker, F. G. Hall, R. B. Myneni, Y. Wang, J. Bi, Y. M. de Moura, and P. J. Sellers, “Vegetation dynamics and rainfall sensitivity of the Amazon,” *Proceedings of the National Academy of Sciences*, vol. 111, no. 45, pp. 16 041–16 046, 2014.

[6] M. Stafoggia, J. Schwartz, C. Badaloni, T. Bellander, E. Alessandrini, G. Cattani, F. DeDonato, A. Gaeta, G. Leone, A. Lyapustin et al., “Estimation of daily pm10 concentrations in italy (2006–2012) using finely resolved satellite data, land use variables and meteorology,” *Environment international*, vol. 99, pp. 234–244, 2017.

[7] A. Mhawish, T. Banerjee, M. Sorek-Hamer, A. Lyapustin, D. M. Broady, and R. Chattfield, “Comparison and evaluation of modis multi-angle implementation of atmospheric correction (maiac) aerosol product over south asia,” *Remote Sensing of Environment*, vol. 224, pp. 12–28, 2019.

[8] H. Jethva, O. Torres, and Y. Yoshida, “Accuracy assessment of MODIS land aerosol optical thickness algorithms using AERONET measurements over North America.” *Atmospheric Measurement Techniques*, vol. 12, no. 8, 2019.

[9] X. Yi, J. Zhang, Z. Wang, T. Li, and Y. Zheng, “Deep distributed fusion network for air quality prediction,” in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 965–973.

[10] D. Qin, J. Yu, G. Zou, R. Yong, Q. Zhao, and B. Zhang, “A novel combined prediction scheme based on CNN and LSTM for urban PM$_{2.5}$ concentration,” *IEEE Access*, vol. 7, pp. 20 050–20 059, 2019.

[11] Y. Zheng, X. Yi, M. Li, R. Li, Z. Shai, E. Chang, and T. Li, “Forecasting fine-grained air quality based on big data,” in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 2267–2276.

[12] X. Meng and G. E. Karniadakis, “A composite neural network that learns from multi-fidelity data: Application to function approximation and inverse PDE problems,” *Journal of Computational Physics*, vol. 401, p. 109020, 2020.

[13] M.-C. Yang and M. C. Chen, “Composite Neural Network: Theory and Application to PM$_{2.5}$ Prediction,” *arXiv preprint arXiv:1910.09739*, 2019.
APPENDIX A

DOWNSCALE SATELLITE TILES

We show how we downscale each satellite tiles in the remote area from 1200x1200 km\(^2\) to 300x300 km\(^2\). The max pooling of 2x2 kernel with 2x2 stride is used to downscale satellite tiles while maintaining the distribution of values like the original tile. The kernel window slides on the satellite image and summarizes each sub-region with the maximum value. Fig.1 shows how max pooling works on satellite tiles with the following shape [hours, height, width], where hours represent all hours in winter and autumn season.

APPENDIX B
CLASSIFICATION DETAILS

The numerical prediction score of PM\(_{2.5}\) is converted into decision by checking if the score is above the defined threshold values. Base on the International Organization for Standardization (ISO) 5725, numeric accuracy is the decomposition of numerical quantities into numerical version of trueness and precision. Therefore, we calculate the score of the evaluation metrics including accuracy, precision, recall and F1-score between prediction results and true values using three defined thresholds of PM\(_{2.5}\). We use the confusion matrix as a tool to calculate those metrics, the confusion matrix contains all the information that can be used to analyze the errors and confusion of the end results [31]. Furthermore, the confusion matrix consists of true positive (TP) information which in this work we define as predicted PM\(_{2.5}\) value which satisfy both thresholds condition. The matrix also contains true negative (TN), false positive (FP) and false negative (FN) information. Table 1 shows the details of the remote transportation event confusion matrix that match with this task. The equations below show how to calculate all classification metrics.

Given PM\(_{2.5}\) ground truth(GT) values and their predicted values as remote transportation (RT) with their corresponding first order differential vector GTD and RTD. Moreover given the $E_{\text{Pepa\_tshd}}(\beta_1)$, $D_{\text{Tiff\_tshd}}(\beta_2)$ and the total number of hours that used in prediction(all hours in winter and autumn seasons). We obtain four events(E) after applying those thresholds to PM\(_{2.5}\) values.
Satellite tiles shape: 8760, 1200, 1200  Maxpool kernel size: 2x2  8760, 600, 600  Maxpool kernel size: 2x2  Downscale Satellite tiles: 8760, 300, 300

Fig. 12: Downscale of Satellite tile from spatial dimension of 1200x1200 to 300x300 using Maximum pooling with kernel size of 2x2

| 2  | 3  | 5  | 1  | 3  | 4  |
|----|----|----|----|----|----|
| 6  | 5  | 4  | 8  | 5  | 5  |
| 3  | 7  | 2  | 1  | 4  | 3  |
| 1  | 2  | 6  | 2  | 6  | 9  |
| 4  | 4  | 3  | 5  | 2  | 4  |
| 7  | 1  | 5  | 0  | 1  | 7  |

Input: 6x6  Kernel size : 2x2, stride: 2x2  Output: 3x3

Fig. 13: Illustration of the max pooling with kernel of 2x2 applied to input image with shape 6x6

E1: GT > β1, E2: GTD > β2, E3: RT > β1, E4: RTD > β2

Then we use probability(P) to obtain the number of remote transportation pollution events by considering their occurrence with total number of hours(t). Later, the confusion matrix in Table I shows how to calculate true positive(TP), false negative(FN), false positive(FP), and true negative(TN), which show the full picture of our model performance.

\[ P(GT) = \frac{\text{count}\{E_1\}}{\text{total}\{t\}} \quad (10) \]

\[ P(GT') = \frac{\text{count}\{E_2\}}{\text{total}\{t\}} \quad (11) \]

\[ P(GT|GT') = \frac{\text{count}\{E_1 \cap E_2\}}{E_2} \quad (12) \]

\[ P(GT \cap GT') = \frac{\text{count}\{E_1 \cap E_2\}}{\text{total}\{t\}} = P(GT|GT') P(GT') \quad (13) \]

\[ P(RT) = \frac{\text{count}\{E_3\}}{\text{total}\{t\}} \quad (14) \]

\[ P(RT') = \frac{\text{count}\{E_4\}}{\text{total}\{t\}} \quad (15) \]

\[ P(RT|RT') = \frac{\text{count}\{E_3 \cap E_4\}}{E_4} \quad (16) \]

\[ P(RT \cap RT') = \frac{\text{count}\{E_3 \cap E_4\}}{\text{total}\{t\}} = P(RT|RT') P(RT') \quad (17) \]

| Ground Truth PM2.5 | Remote event=Yes | Remote event=No |
|--------------------|------------------|-----------------|
| Remote Event=Yes   | TP= (GT ∩ GT') ∩ (RT ∩ RT') | FN= (GT ∩ GT') ∩ ¬(RT ∩ RT') |
| Remote Event=No    | FP= ¬(GT ∩ GT') ∩ (RT ∩ RT') | TN= ¬(GT ∩ GT') ∩ ¬(RT ∩ RT') |

George William Kibirige received the B.Sc. degree in Information and Communication Technology Management from Mzumbe University, Tanzania and the MSc. Computer Science (Information Security) from University of Technology Malaysia, Malaysia. He is currently pursuing the Ph.D. degree in Air pollution with the National Chengchi University in Taiwan.

His research interests include machine learning, deep learning, data analysis, and their applications in air pollution and social networks.

Ming-Chuan Yang received the BS and MS degrees in mathematics from National Tsing Hua University, Taiwan, and the Ph.D. degree in Computer Science and Engineering from National Chiao Tung University, Taiwan, in 2015. Currently, he is a postdoctoral fellow of the Institute of Information Science, Academia Sinica, Taiwan.

His research interests include deep/machine learning theory, discrete mathematics, algorithms, and computational complexity.
Chao-Lin Liu is a distinguished professor of the National Chengchi University in Taiwan. He received a doctorate in intelligent systems from the University of Michigan, in USA. His research interests include probabilistic reasoning, machine learning, natural language processing, and digital humanities.

Meng Chang Chen received the Ph.D. degree in Computer Science from the University of California, Los Angeles, in 1989. He was with AT&T Bell Labs from 1989 to 1992 and now he is a Research Fellow/Professor of Institute of Information Science, Academia Sinica, Taiwan. His current research interests include computer and network security, wireless network, deep learning for complicated applications, data and knowledge engineering.