

Airstacknet: A Stacking Ensemble-Based Approach for Air Quality Prediction

Amel Ksibi¹, Amina Salhi¹, Ala Saleh Alluhaidan^{1,*} and Sahar A. El-Rahman²

¹Department of Information Systems, College of Computer and Information Sciences,
Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia

²Electrical Engineering Department, Faculty of Engineering-Shoubra, Benha University, Cairo, Egypt

*Corresponding Author: Ala Saleh Alluhaidan. Email: asalluhaidan@pnu.edu.sa

Received: 22 May 2022; Accepted: 23 June 2022

Abstract: The quality of the air we breathe during the courses of our daily lives has a significant impact on our health and well-being as individuals. Unfortunately, personal air quality measurement remains challenging. In this study, we investigate the use of first-person photos for the prediction of air quality. The main idea is to harness the power of a generalized stacking approach and the importance of haze features extracted from first-person images to create an efficient new stacking model called AirStackNet for air pollution prediction. AirStackNet consists of two layers and four regression models, where the first layer generates meta-data from Light Gradient Boosting Machine (Light-GBM), Extreme Gradient Boosting Regression (XGBoost) and CatBoost Regression (CatBoost), whereas the second layer computes the final prediction from the meta-data of the first layer using Extra Tree Regression (ET). The performance of the proposed AirStackNet model is validated using public Personal Air Quality Dataset (PAQD). Our experiments are evaluated using Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Coefficient of Determination (R^2), Mean Squared Error (MSE), Root Mean Squared Logarithmic Error (RMSLE), and Mean Absolute Percentage Error (MAPE). Experimental Results indicate that the proposed AirStackNet model not only can effectively improve air pollution prediction performance by overcoming the Bias-Variance tradeoff, but also outperforms baseline and state of the art models.

Keywords: Personal air quality; prediction; airstacknet; ensemble learning; feature extraction; stacking

1 Introduction

Air pollution is one of the greatest environmental threats to human health and well-being worldwide. According to the World Health Organization (WHO), exposure to air pollution leads to nearly seven million premature deaths each year due to the exacerbation of many chronic diseases such as respiratory infections, aggravated asthma, heart disease, and strokes [1,2]. Monitoring air pollution is therefore of paramount importance. It is necessary to study the characteristics of the surrounding



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

environment and track air pollution accurately so that people can be aware of the air pollution levels in their current location and accordingly act.

To quantify the level of air pollution, government agencies have defined the Air Quality Index (AQI). AQI is conventionally derived by sensing, sampling, and measuring, regularly throughout the city, the microgram per cubic meter (g/m^3) concentrations of several air pollutants, including particulate matter ($\text{PM}_{2.5}$), ozone (O_3), and nitrogen dioxide (NO_2) [3,4]. This method has severe limitations in terms of time and location, due to variability in AQI within a city related to weather, traffic, and land use [5]. Accordingly, people should have access to real-time pollution concentration information as they commute to work, walk around town, or spend time outside, so they can choose the healthiest route to their destination and take precautions to protect their health [3,6]. Hence, determining the personal AQI accurately in real-time is a topic that merits further exploration. Accordingly, local information on air pollution (e.g., $\text{PM}_{2.5}$, NO_2 , O_3) and weather (e.g., temperature and humidity) is important at the personal level, however, it is not always possible to collect a large amount of these data [7,8]. Rather than relying solely on conventional air monitoring stations, mobile technology and artificial intelligence can provide valuable clues from first-person images to understand that environmental situations in cases in which precise data from air pollution stations are lacking. In fact, image quality has steadily improved, allowing for the estimation of AQI from images using image processing and machine learning [5,7,8]. Thus, the public can easily take photos of the surrounding environment using their mobile phones, get instant air quality information from the image analysis process, and take timely actions to prevent exposure to air pollution.

Recently, the prediction of air quality based on the analysis of images has received increasing attention. The assumption behind image-based air quality prediction is the correlation between environmental data ($\text{PM}_{2.5}$, O_3 and NO_2 data as ground truth) and images. It enables the development of reliable image-based models that can provide insights into personal AQI. These models use image processing algorithms to extract features from first-person images and estimate the air quality based on these features [5,9–11]. Despite the above Image-based prediction models having acceptable prediction performances, the use of a single model for prediction to deal with the randomness of air pollution concentrations often leads to poor model generalization [12]. Using a single model is unlikely to capture the entire underlying structure of the data to yield optimal results. A model with a high variance may represent the data set accurately, but it may also be over-fitted to noisy or unrepresentative training data. Likewise, a model that exhibits a low variance and a large bias will underfit the target. As an alternative, Ensemble learning, as a general meta-approach to machine learning, has become a hot research topic. The method seeks to make better predictions by combining multiple learners through an ensemble of models in order to capture the underlying distribution of the data more accurately [13,14].

Boosting, Bagging, and Stacking are the three most prevalent Ensemble learning approaches. The advantage of stacking is that it can combine the capabilities of several heterogeneous high-performing learners to solve a regression or classification problem, whereas bagging and boosting consider only homogeneous weak learners. It is worth mentioning that bagging and boosting focus more on reducing the variance and bias, respectively, while stacking targets both by finding the model that exhibits a small variance and a large bias.

Stacking, also known as a stacked generalization, consists of combining multiple learners where a meta-level (or level-1) learner uses the output of base-level (or level-0) learners. This approach has been introduced to the field of air pollution prediction and shows improvements in the prediction accuracy [13,14]. Most of these applied stacking models predict and forecast air quality based on

weather, location, and environmental sensors data [15,16]. Furthermore, these models use arbitrary choices of Level-0 and Level-1 learners without focusing on optimizing the stacked structure.

To the best of our knowledge, this is the first study that deals with the optimal combination of base-level and meta-level learners, to predict $PM_{2.5}$, O_3 , and NO_2 concentrations, at a precise place, using discriminate haze-relevant features extracted from photos rather than using environmental data. Indeed, we define a novel optimized Stacking model, AirStackNet, to improve the overall regression performance. AirStackNet consists of Boosting algorithms as the base-level learners (XGBoost, CatBoost, LightGBM) and one of Bagging algorithms (Random Forest/Extra-trees) as the meta-level learner. The greatest strength of AirStackNet is that the meta-data inherited from its diverse boosting base-level learners enhance its training performance to generate a better-quality model. Our model aims to compensate for the inadequacies of current air quality detection technologies and provide a fine-grained, low-cost air quality monitoring. To evaluate the designed model, we perform experiments on Personal Air Quality Data Set (PAQD) to demonstrate the generalizability of the designed stacking structure.

The main research questions that we want to address in this paper are:

- 1) How should base-level and meta-level learners be paired in AirStackNet?
- 2) Can the AirStackNet stacking structure outperform the baseline learners?
- 3) What effect does the tuning of the hyperparameters of the base-level learners have on AirStackNet's prediction quality?
- 4) What is the performance of AirStackNet in comparison to other ensemble techniques?

Our paper is organized as follows: In Section 2, we provide an overview of air quality prediction methods. In Section 3, we describe our proposed model, AirStackNet, for air quality prediction. In Section 4, we present the experiments and discussion.

2 Related Work

Personal air quality is a significant indicator during the evaluation of the air pollution impact on personal health [17]. Air Quality Index (AQI) is a rank indicator of air pollution, which is a very critical environmental impact on the public health [10]. Over the last 40 years, city-wide air quality prediction has been of interest [18]. All of these studies, however, were limited to evaluating the levels of air pollutants at the city scale for the general population. At the individual level, current research has focused on crowdsourcing computing by capturing data from sensing devices [8]. These sensors produce lifelog data that can be categorized into numerical data and non-numerical data (environmental variables, weather data, Global Positioning System (GPS), health measurements, time, etc). In recent years, the air quality measurement based on computing methods using lifelogging data has attracted much attention of the researchers [11]. Predicting personal air quality has the main challenge that is developing an effective model based on a small training dataset in contrast to the public stations of atmospheric monitoring [17]. This work aims to predict personal air quality utilizing numerical lifelog data.

Reference [15] studied a methodology based on multi-source machine learning to estimate the local AQI score and level according to the location of users in a huge city. The utilized machine learning algorithms are Extreme Gradient Boosting, CatBoost, LightGBM, Random Forest, and Support Vector Machine. The researchers conducted several experiments on three main datasets are "MNR-HCM", "MNR-Air-HCM", and Surrounding-Environment Personal-Health Lifelog Archive (SEPHLA)-MediaEval 2019 gathered in Fukuoka (Japan) and Ho Chi Minh (Vietnam) cities. Various

useful related features are extracted from the datasets, such as geographic data, timestamp information, sensor data (temperature and humidity), emotion tags from users (such as calm, green, etc.), the semantic attributes of images taken by users, and the public weather data (including pressure, temperature, humidity, dew point, and wind speed) of the corresponding cities. The results showed that the random forest model with sensor data consolidated with public weather data is best suited to estimate AQI values and ranks in many experiments.

Reference [19] presented a study to predict AQI level based on several models of machine learning using sensor data (including temperature, humidity), location, timestamp features, and public weather data. The experiments are conducted using “MNR-HCM II” dataset. The findings found that adopting stacking generalization with these features achieves higher overall performance than other models and features.

Reference [20] developed different predicting models based on machine learning algorithms to predict AQI level utilizing an eleven-year dataset gathered by Taiwan’s EPA (Environmental Protection Administration). The utilized machine learning algorithms are stacking ensemble, artificial neural network, adaptive boosting, random forest, and support vector machine. The findings found promising results for predicting AQI level and the best performance found in the stacking ensemble and AdaBoost.

Reference [21] proposed a framework of aerial-ground air quality sensing for fine-grained three dimensional AQI monitoring and predicting based on a light-weight Dense-MobileNet model using haze images acquired by UAVs (Unmanned Aerial Vehicles). Also, the researchers presented a Graph Convolutional neural network-based Long Short-Term Memory (GC-LSTM) model using graph topology to realize AQI prediction. The experiments are conducted using a real-world dataset.

Reference [22] developed a simple but effective image prior that is a dark channel prior for haze removal from a single image. The researchers directly estimated the haze thickness and recover a high-quality image of haze-free. Reference [23] proposed an approach based on Support Vector Machine (SVM) and Random Forest (RF) models to estimate the local AQI level and score. The experiments are conducted using the dataset provided at MediaEval 2020 based on K-Fold cross-validation and Randomized-Search.

The researchers in [24] developed a model to forecast the personal AQI using IDW (Inverse Distance Weighted) algorithm for estimating the missing levels of air pollutants. The experiments are conducted using the dataset provided at MediaEval 2020 based on the pollutants O_3 , NO_2 , $PM_{2.5}$. In [7] presented a voting regression algorithm to predict AQI using three base regressors that are Random Forest, Gradient Boosting, and linear regressor. The experiments are conducted using the dataset provided at MediaEval 2020. Reference [25] presented a complex event analysis to predict the traffic risk based on three Dimensional-Convolutional Neural Network (3D-CNN) and a related events set that are extracted from different sources of urban sensing data. Where they a model to reserve and wrap the spatio-temporal information into 3D raster images to conduct the predictive analytics using 3D-CNN. The researchers evaluated the proposed model by the real dataset during 2014 and 2015 collected in Kobe, Japan.

In [26], the authors proposed a prediction model based on CRNN (convolution recurrent neural network) for short-term $PM_{2.5}$ pollution prediction utilizing the spatial-temporal features of atmospheric sensing data. The experiments were conducted using the atmospheric sensing dataset from thirty-three coastal cities in China and Fukuoka’s environmental monitoring dataset from 2015 to 2017.

In [11] developed a DCWCN (double-channel weighted convolutional network) ensemble learning method utilizing the extraction features from various parts of the environmental image. For feature extraction, the researchers built a DCWCN using each channel for training the various parts of the environment images. Then a self-learning weighted feature fusion method was proposed that weights and concatenates the vectors of extracted features for measuring the air quality. The experiments were conducted using a dataset of an environmental image with random sampling time and locations.

In [17], the authors developed a transfer learning model using an encoder-decoder structure based on Wasserstein distance to match atmospheric monitoring station data representing the heterogeneous distribution of the source domain and personal air quality representing the target domain. The experiments were conducted using the dataset of public atmospheric monitoring stations collected by AEROS (Atmospheric Environmental Regional Observation System) in Japan as the source domain. The target domains are the private data collected in Fujisawa and Tokyo, Japan.

In [27], the authors analyzed the association among the greenness exposure and the depressive symptoms based on constructing the logistic regression models. The experiments are conducted based on the data collected in 2009 from the Korean Community Health Survey. NDVI (Normalized Difference Vegetation Index) and land-use data (forest area and forest volume) are utilized to assess the greenness. To report the depressive symptoms, the OR (Odds ratios), with 95 percent confidence intervals (CI) were estimated with the quartile 1 (quartile of the lowest NDVI).

In [28] presented two frameworks to estimate the air pollution level. In the first framework, the images were preprocessed, the features extracted from the images using Gabor transform, and then two shallow classifiers are utilized for modeling and predicting the air pollution level. In the second framework, CNN was developed to input a sky image and output the air pollution level. The experimental findings found that CNN classifier fulfilled accuracy better than the traditional combination methods of feature extraction and classification. In [10] presented an association framework between AQI rank of lifelogging data and visual features. In the training phase, the visual data (environmental images) and lifelog numerical data (environmental AQI measurements) are used. CNN model is used for the feature extraction of the visual data and the standard AQI ranking is used for numerical data. Then the outputs are combined as the input data for a multi-layer perception algorithm to estimate the relationship of association between the visual feature and the rank of AQI.

The researchers in [13] developed a stacked ensemble model to analyze and forecast the daily average concentration of $PM_{2.5}$ in Beijing city (China). Special processes are conducted to extract significant features before modeling, including simplification, transformation polynomial, and combination. Tree-based and stability feature selection approaches are applied to determine the important attributes. XGBoost, Adaboost, Least Absolute Shrinkage and Selection Operator (LASSO), and multi-layer perceptron with the genetic algorithm are utilized in the space of level-0 and then in the space of level-1 are integrated by support vector regression via stacked generalization. The experiments indicated that the ensemble model produces better performance than one non-linear predicting model. The summary of existing approaches is shown in [Tab. 1](#).

Table 1: Summary of existing approaches

Ref.	Techniques	Dataset	Performance measurements	Limitations
[21]	Graph convolutional neural network-based Long Short-term memory (GC-LSTM)	Real-world dataset	-GC-LSTM: RMSE up to 9.145	-Expensive communication overhead. -Difficulty in achieving long-term AQI monitoring. -Large-scale and complicated deep learning models cannot be deployed to UAVs.
[6]	Semantic and instance segmentation, Image retrieval, Transfer learning, Fuzzy negation and Periodic frequent pattern mining	MNR-HCM, VisionAir India	-Prediction Precision: MNR-HCM = 0.9022 and VisionAir India 0.7343.	-location constraints. -No estimation of $PM_{2.5}$ at the Real time.
[5]	A deep learning and image-based model using residual neural network (RESNET)	(Northwest Normal University dataset) NWNNU-AQI: multi-scenario air quality image database	-Mean Accuracy = 74	-Daytime air quality monitoring. -Due to different constraints such as (atmospheric differences and different regional climate), the model will not be able to match the monitoring accuracy of air quality monitoring stations.

(Continued)

Table 1: Continued

Ref.	Techniques	Dataset	Performance measurements	Limitations
[4]	Semisupervised learning, Artificial neural network and Linear-chain conditional random field	Road networks, taxi trajectories, meteorological data and point of interests (POIs)	-PM ₁₀ : Precision = 0.828 and Recall = 0.826. -NO ₂ : Precision = 0.808 and Recall = 0.798.	-Datasets variety. -There are only a few existing monitor stations. -Aerosols and other pollutants are not included in the methodology.
[3]	AQI-T-RM (Air pollution and Traffic Risk Map)	MNR-HCM	-Exposure reduction 30.25.	-Data with a granularity of less than 10 m was collected.
[23]	SVM and RF	Dataset provided at mediaeval 2020	-SVM: MAE 12.74, RMSE 15.93, and SMAPE 0.32. -RF: MAE 12.93, RMSE 15.96, and SMAPE 0.32.	-Each subtask's model is optimized and evaluated separately.
[24]	IDW	Dataset provided at MediaEval 2020	-SMAPE: Up to 0.521219253 -RMSE: Up to 78.4465017 -MAE: Up to 30.32401094	-There is a significant gap in O ₃ and NO ₂ prediction results. Because the model equation does not have any offset factors to adjust for the large difference between personal weather data collected by volunteers and public weather data from weather stations.

(Continued)

Table 1: Continued

Ref.	Techniques	Dataset	Performance measurements	Limitations
[7]	Voting regression algorithm based on three base re-gressors that are Random Forest, Gradient Boosting, and linear regressor	Dataset provided at MediaEval 2020	-RMSE: average score of 35.39 for Walker and 51.16 for Car.	-AQI and individual air pollution levels are predicted based on regression methods.
[11]	DCWCN (double-channel weighted convolutional network)	Building an environmental image database with random locations and sampling time.	-Accuracy over 87%	-It's difficult to discriminate between images with adjacent grades and similar information.
[25]	3D-CNN	Real dataset collected in Kobe, Japan	-Short-term: Best MSE is 5494.44 -Medium-term: Best MSE is 6676.25 -Long-term: Best MSE is 6617.74	-Prediction model based on atmospheric environmental data

3 Methodology

This section describes the proposed approach and its associated process flow that consists of three main steps for predicting personal air quality index, as shown in Fig. 1. The main process flow is as follows:

Step 1: Data preparation.

Missing values can heavily influence regression models. Personal Air Quality Data Set (PAQD) contains some missing values such as air pollutants concentration data $PM_{2.5}$, NO_2 , O_3 and weather data (e.g., temperature and wind). We apply Imputation method using Ball Tree algorithm with Haversine distance to estimate the missing values.

Step 2: Haze-relevant Features Extraction.

224-dimension vectors of six Haze-relevant features were obtained from original images.

Step 3: Model construction.

The extracted feature vectors were entered into the base regressors which are XGBoost, LightGBM, and CatBoost to estimate the probability of each of pollutant concentration from each base regressor. The obtained vector of meta-data from the base-level learners was entered into the Random Forest/Extra-Trees meta-regressor to predict $PM_{2.5}$, O_3 , and NO_2 values from images.

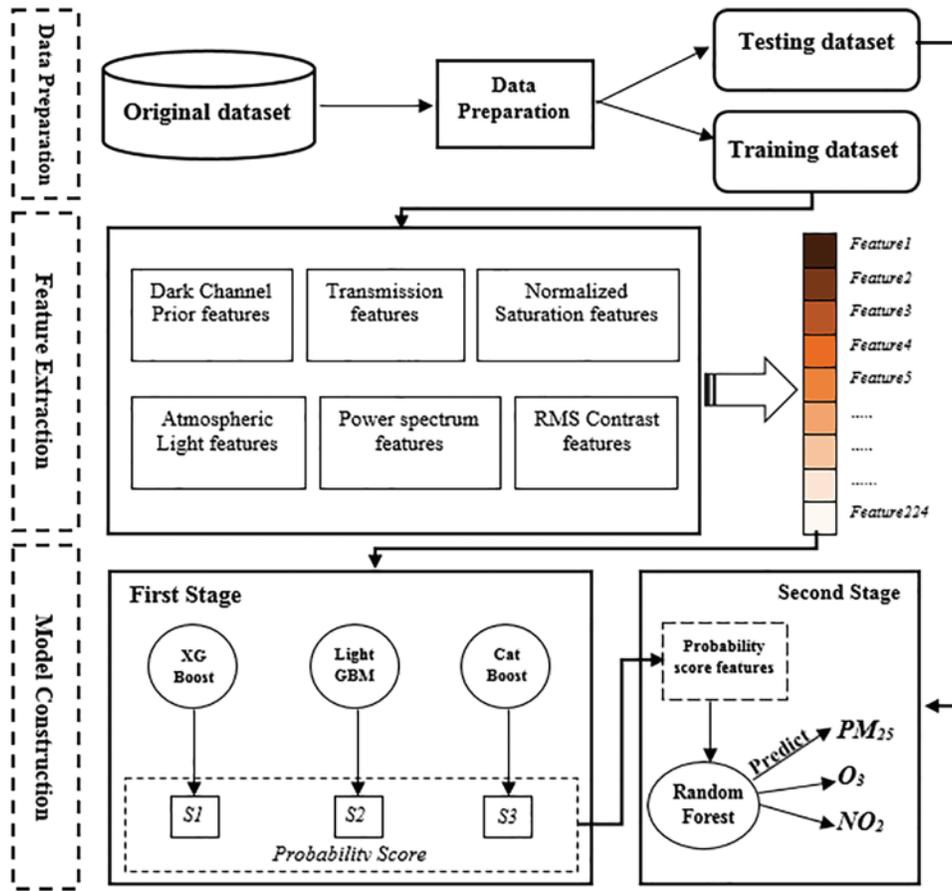


Figure 1: System architecture

3.1 Data Preparation

To impute the missing data, we propose to perform a local neighborhood search using a BallTree k-nearest neighbors (KNN) algorithm with the Haversine distance metric. The missing values of the pollutants are replaced by the mean of the known values in the nearest neighboring areas. Haversine, as a formula for navigation, calculates the angular distance between two points on the surface of a sphere. This metric calculates the shortest path between two points, just like Euclidean distance. The difference is that a straight line is not possible since both points are assumed to be on a sphere. To calculate Haversine distance between two points on the sphere, The first coordinate of each point is assumed to be the latitude, the second is the longitude, given in radian [29]:

$$D(x, y) = 2arcsin \left[\sqrt{\sin^2 \left(\frac{x1 - y1}{2} \right) + \cos \cos (x1) * \cos \cos (y1) \sin^2 \left(\frac{x2 - y2}{2} \right)} \right] \quad (1)$$

3.2 Haze-relevant Features Extraction

Haze is an atmospheric phenomenon where dust, smoke, and other fine particles obscure the clarity of the scene. Therefore, outdoor photographs can be used to estimate the concentration of air pollutants using haze information. We need to find the statistical features that are most relevant to the

haze density in the image, regardless of the image content. In image processing, an image influenced by haze can be described using the atmospheric model follows [30]:

$$I(x) = J(x) t(x) + A(1 - t(x)) \quad (2)$$

where x is the pixel coordinates, I is the observed hazy image, J is the haze-free image, t represents the medium transmission, and A denotes the global atmospheric light. Images with higher air pollution tend to look hazier due to lower transmission and contrast. Hence, image features that correlate to haze level enable pollutant concentration estimation. In the following, we investigate six haze-relevant features.

- a. **Dark Channel Prior:** Dark channel prior is based on the observation that in most haze-free outdoor images at least one color channel has some pixels with very low intensity. Conversely, hazy images have a significant increase in luminance as a result of additive air light, resulting in no dark pixels. The dark channel is an informatocurreive feature for haze detection. It is defined as the minimum of all pixel colors in a local area and can be calculated using the following equation [31]:

$$J_{dark}(x) = \min_{c \in r, g, b} \left(\min_{y \in \Omega_r(x)} J^c(y) \right) \quad (3)$$

where J^c represents one of the RGB channel of J , and x denotes all pixel colors in a local patch.

- b. **The Transmission:** Based on the Dark Channel Prior, we assume that the atmospheric light A is given, and the transmission is constant in a local area $r(x)$. If we use the minimum operation in the local area in Eq. (1), we can estimate the transmission as follows:

$$\underline{t} = 1 - \omega \min_c \left(\frac{\min_{y \in \Omega_r(x)} I^c(y)}{A^c} \right) \quad (4)$$

In our experiment, we fix the value of w to 0.95, the patch is set to 45×45 for an image.

- c. **RMS Contrast:** Low contrast is one of the observable characteristics of a blurred image due to the scattering and diffusion of reflected light reaching the cameras. Therefore, contrast is one of the most perceived features for haze detection and estimation. In this study, we use the root mean square (RMS) of an image to describe the contrast, which is defined as the standard deviation of the image pixel intensities [21].

$$RMS = \sqrt{\left(\frac{1}{mn} \left(\sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \left(I_{ij} - \text{avg}(I) \right)^2 \right) \right)} \quad (5)$$

where I^i is intensity at (i, j) pixel of the image with size M by N , and $\text{avg}(I)$ is the average intensity of all pixels in the image.

- d. **Atmospheric Light:** To estimate the atmospheric light, we first pick the top 0.1% brightest pixels in the dark channel. Among these pixels, the pixels with highest intensity in the input image I is selected as the atmospheric light [22].
- e. **Power Spectrum:** Usually, air pollution time series have a broad spectrum related to the periodicity of physical processes in the atmosphere and precursor emissions. We calculate the power spectrum of an image I with magnitude NN by squaring the magnitude according to the Discrete Fourier Transform (DFT) [32]:

$$S(u, v) = 1/N^2 |I(u, v)|^2 \quad (6)$$

where (u, v) denotes the image transformed by DFT. We represent the two-dimensional frequency in polar coordinates, thus $u = f \cos \theta$ and $v = f \sin \theta$, f denotes the radius of power spectrum image, and θ denotes the angle of the polar coordinates.

- f. **Normalized Saturation:** It can be observed that the image saturation varies greatly with the change in haze concentration. Therefore, we consider the characteristic of local saturation, which indicates how close the colour is to the spectral colour [33]. For an image I , we calculate the normalized saturation for each pixel by:

$$S_{x,y} = (I_{x,y} - \min(S_1)) / (\max(S_1) - \min(S_1)) \quad (7)$$

where $\max(S_1)$ is the maximum saturation value and $\min(S_1)$ is the minimum saturation value of image I .

3.3 Stacked Ensemble Regressor and Model Construction

Single-base regressors are algorithms that follow the basic rules of machine learning. These algorithms use a training dataset and apply only one of the machine learning algorithms to build a predictive model, e.g., LR, MLP, and SVM, etc. Ensemble methods, on the other hand, usually combine many weak learning algorithms to obtain a strong model. The stacked ensemble regressor is an integrated method proposed by [34]. The prediction results of several ordinary learners are used as new features for retraining to achieve a minimum error rate of the prediction model. Compared to the individual regressors, the stacking regressor has a stronger non-linear expressiveness. In this paper, we use the AirStackNet model to predict personal air quality, as presented in Algorithm 1. The AirStackNet model is based on a stacked ensemble regressor that includes two learning stages to predict pollutant concentration levels. In the first stage, the extracted visual features are input to the base-level boosting regressors. We decide to use the combination of XGBoost, LightGBM and CatBoost as boosting base regressors. Then, the output of the base regressors are estimated via ten-fold cross-validation to avoid the problem of over-fitting. In the second stage, these outputs are used as input for the bagging meta-regressor (Random Forest/Extra trees regressors).

The idea of cross validation is combined with Stacking to avoid using the same training set for building both base-level and meta-level regressors. Instead of using all the training examples to get base-level regressors, we now partition data in D into K subsets and get K regressors, each of each is trained only on $K-1$ subsets. Each regressor is applied to the remaining one subset and the output of all the base-level regressors constitute the input feature space for the meta-level regressor. After we build the meta-level regressor based on the base-level regressors's predicted labels, we re-train the base-level regressors on the whole training set D so that all the training examples are used. Applying the meta-level regressor on the updated base-level regressors outputs provides the final ensemble output.

Algorithm 1: AirStackNet with K-fold Cross Validation

Data: Training data $D = (x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)$

Step 1: split D into K subsets: $D = \{D_1, D_2, \dots, D_K\}$

for k in $[1..K]$ **do**

Learn T base-level regressors

for t in $[1..T]$ **do**

 train a regressor h_{kt} from $D \setminus D_k$

end for

(Continued)

Algorithm 1: Continued

 Construct a training set for the meta-level regressor

```

for  $x_i \in D_k$  do
  calculate  $\{x'_i, y_i\}$ ,
  where  $x'_i = \{hk1(x_i), hk2(x_i), \dots, hkT(x_i)\}$ 
end for

```

```

end for

```

Step 2: Learn a meta-level regressor

Learn a new regressor h' from the set of $\{x'_i, y_i\}$

Step 3: Re-train the base-level regressors

```

for  $t$  in  $[1..T]$  do

```

Train a regressor h_t based on D

```

end for

```

Result: $H(x) = h'(h_1(x), h_2(x), \dots, h_T(x))$ where H is an ensemble regressor

4 Experimental and Results

The aim of this section is to present key aspects of the dataset, the model settings, and the performance evaluation metrics we used to compare the proposed stacked ensemble model with other selected models. All the experiments conducted to answer the above-mentioned research questions are summarized in this section.

4.1 Dataset Description

To conduct the experiments, we used the Personal Air Quality Data Set (PAQD), provided within the MediaEval 2020 task “Multimodal personal health lifelog data analysis”, to answer the following question “Can personal air quality be predicted using images captured by personal devices and some open source data?” [8]. The aim of this task is to find out if we can use only lifelog data (i.e., images of the environment) and some open source data (such as weather and air pollution data) to predict personal air pollution data. PAQD was collected from March to April 2019 along the marathon route at the 2020 Tokyo Olympics and the running route around the Imperial Palace using wearable sensors. There were five data collection participants who were assigned to five routes to collect the data. Each participant started collecting data at 9AM on each day of the week and took approximately one hour to complete each route. The data collected includes weather data (e.g., temperature and humidity), atmospheric data (e.g., O_3 , $PM_{2.5}$, NO_2 , GPS data, and captured photos. The dataset contains 116751 records and their corresponding images. Each record contains the following information: ID of the record, Identifier of the user, DateTime, Latitude, Longitude, Altitude, Wind speed, PM_1 , $PM_{2.5}$, PM_{10} , O_3 , NO_2 and the path to the associated photo.

In order to gain insights from raw data that helps in making informed decisions, we performed descriptive statistical data analysis. We calculated the mean, min, std, 25%, 50%, 75% and max values for each of the atmospheric data (e.g., O_3 , $PM_{2.5}$, NO_2 . Fig. 2 summarizes the obtained results. Pollution levels in central Tokyo remain high, as evidenced by a maximum $PM_{2.5}$ level of 46 and a mean value of 22, which is categorized as unhealthy for sensitive groups.

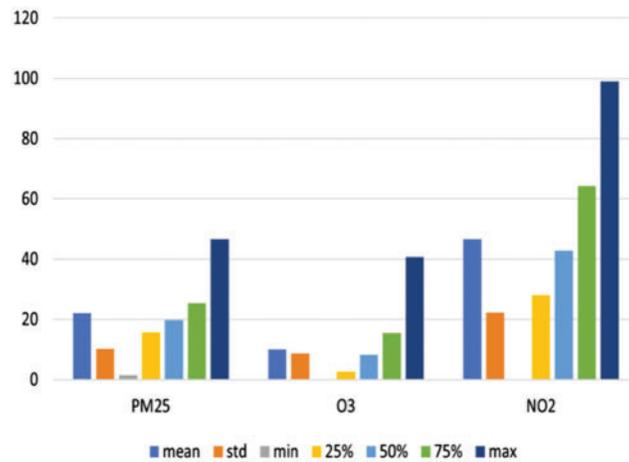


Figure 2: “Personal Air Quality Dataset” (PAQD) analysis

4.2 Models Setting

Our models are implemented on Windows™ 10 Pro, Intel(R) Core(TM) i7-10510U CPU@ 1.80 GHz 2.30 GHz, with RAM storage of 16 GB. We used the regression class from the PyCaret library, an open source low-code machine learning library in Python. During the training process, 70% of the images were randomly selected for the training set, and the remaining 30% were used as the test set. During the training process, we tuned the hyperparameters using Randomized SearchCV with 10-fold cross-validation to find the best combination of hyperparameters for each base regressor from the list of base regressors (Gradient Boosting Regressor, LightGBM Regressor, CatBoost Regressor). Randomised SearchCV is a very effective method for tuning the parameters that increase the generalizability of the model. This process searches a predefined parameter space for each model and selects the hyperparameters with the best performance. After 50 iterations of the random search process via 10-fold cross-validation to optimize the performance metric “MAE”, we obtained the optimal regressor parameters. The best performing hyper-parameters selected based on the lowest values on MAE are used to fit each regressor to the full training dataset. [Tab. 2](#) shows the results of tuning the hyper-parameters for each base regressor. The tuned regressors were used as base-level regressors to predict the concentration values of the following pollutants (NO_2 , O_3 , $\text{PM}_{2.5}$).

Table 2: Details of hyperparameters tuned values for each regression model

Regressor	Tuned hyper-parameters values
LightGBM regressor	Bagging_fraction = 0.8, Bagging_freq = 3, Boosting_type = ‘gbdt’, Feature_fraction = 0.6, Importance_type = ‘split’, Learning_rate = 0.15, Max_depth = -1,

(Continued)

Table 2: Continued

Regressor	Tuned hyper-parameters values
XGBoost regressor	Min_child_samples = 36, Min_child_weight = 0.001, Min_split_gain = 0.6, N_estimators = 200, Num_leaves = 256 Base_score = 0.5, Booster = 'gbtree', Colsample_bylevel = 1, Colsample_bynode = 1, Colsample_bytree = 0.7, Enable_categorical = False, Learning_rate = 0.15, Max_delta_step = 0, Max_depth = 11, Min_child_weight = 3, N_estimators = 140, Objective = 'reg:squarederror'
CatBoost regressor	Learning_rate = 0.2, Bagging_temperature = 0.1, l2_leaf_reg = 30, Depth = 12, Max_bin = 255, Iterations = 100, Loss_function = 'Logloss', Bootstrap_type = 'Bayesian'

4.3 Evaluation Metrics

To evaluate the performance of the prediction models for $PM_{2.5}$, NO_2 , and O_3 , we used six evaluation metrics, namely Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Coefficient of Determination (R^2), Mean Squared Error (MSE), Root Mean Squared Logarithmic Error (RMSLE), and Mean Absolute Percentage Error (MAPE), as seen in [Tab. 3](#). MAE represents the average difference between the target value and the predicted value. RMSE represents the square root of the standard deviation of the differences between the predicted and actual values. R^2 is the proportion of the variance in the response variable that is predictable by the explanatory variables. MSE is the second moment of the error and thus includes both the variance of the estimator and its bias. RMSLE is like RMSE but is mainly used for large variance predictions by transforming the predicted and the real dependent variable into a logarithmic value. MAPE means the average difference between the predicted value and the target value. A good regression model is the one that has the lowest MAE, RMSLE, MAPE, MSE, RMSE, and the highest R^2 .

The mathematical formulae are presented in the following equations, where y_i is the actual value of the i^{th} sample case, \hat{y}_i is the predicted value of the i^{th} sample case, \bar{y} is the average of the actual values and n denotes the total number of sample cases.

Table 3: Summary of used performance metrics

Mean absolute error	$MAE = \frac{1}{n} \sum_{i=1}^n y_i - \hat{y}_i $
Mean squared error	$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$
Root mean squared error	$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$
Coefficient of determination	$R^2 = 1 - \frac{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}{\frac{1}{n} \sum_{i=1}^n (y_i - \bar{y})^2}$
Mean absolute percentage error	$MAPE = \frac{100\%}{n} \sum_{i=1}^n \frac{ y_i - \hat{y}_i }{y_i}$
Root mean squared logarithmic error	$RMSLE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\log \log (y_i + 1) - \log \log (\hat{y}_i + 1))^2}$

5 Results and Discussion

To assess the proposed approach and illustrate the performance of the AirStackNet model, we conducted four case studies over the PAQD dataset, in accordance with the research questions. First, we compared the conventional regressors to determine which estimators should serve as base and meta regressors in our AirstackNet model structure. In addition, we studied whether AirstackNet could enhance prediction capabilities compared to conventional individual models. Furthermore, we investigated the impact of tuning the hyperparameters on the reliability of the selected base regressors, yet the stability of AirstackNet. Lastly, in order to assess AirstackNet’s contribution to the literature, we compared it with state-of-the-art models. Below are the details and results of the case studies.

5.1 First Case Study: Choice of Base Regressors

In the first case study, we aim to select the top-10 base estimators for our stacking model. Thus, we conducted comparative study between the following conventional regressors to estimate PM_{2.5}, NO₂, and O₃ pollutants concentration values. The obtained results are reported in [Tabs. 4–6](#).

Table 4: Comparison among different conventional regressors for predicting PM_{2.5} values based on PAQD dataset

Model	MAE	MSE	RMSE	R ²	RMSLE	MAPE	TT (Sec)
Extra trees regressor	0.8018	3.3342	1.8253	0.9679	0.1318	0.0698	71.6620
Random forest regressor	0.9638	5.2981	2.3013	0.9490	0.1453	0.0766	351.2130
CatBoost regressor	2.0008	9.2298	3.0377	0.9110	0.2021	0.1536	27.5680

(Continued)

Table 4: Continued

Model	MAE	MSE	RMSE	R ²	RMSLE	MAPE	TT (Sec)
Extreme gradient boosting	2.0140	9.7609	3.1241	0.9059	0.2005	0.1496	60.5490
Light gradient boosting machine	2.6135	15.7547	3.9688	0.8482	0.2415	0.1984	2.6150
Gradient boosting regressor	4.3352	37.1397	6.0939	0.6421	0.3506	0.3502	116.5250
AdaBoost regressor	6.3794	60.7483	7.7934	0.4145	0.4474	0.5387	23.8170
Orthogonal matching pursuit	6.6966	75.9248	8.7132	0.2683	0.5413	0.7060	0.0570
K Neighbors regressor	6.1283	76.6029	8.7519	0.2617	0.5083	0.5967	0.8120
Ridge regression	6.9717	79.4272	8.9002	0.2346	0.5454	0.7039	0.1750
Elastic net	7.1860	84.3538	9.1842	0.1871	0.5447	0.7162	2.0590
Lasso regression	7.1909	84.4550	9.1897	0.1861	0.5448	0.7163	1.6550
Linear regression	7.2028	84.9466	9.2164	0.1813	0.5459	0.7181	0.3420
Huber regressor	6.8999	94.3894	9.7152	0.0904	0.5515	0.7026	3.8250
Bayesian ridge	7.9878	103.6192	10.0852	0.0014	0.6205	0.7618	1.3990

Table 5: Comparison of base regressors performance before and after hyperparameters tuning

Model	Before tuning			After tuning		
	MAE	R ²	RMSLE	MAE	R ²	RMSLE
Catboost regressor	2.0008	0.9110	0.2021	1.7549	0.9339	0.1817
Extreme gradient boosting	2.0140	0.9059	0.2005	1.0924	0.9588	0.1408
Light gradient boosting machine	2.6135	0.8482	0.2415	1.1070	0.9628	0.1376

Table 6: Comparison of AirStackNet variants performance vs. individual base regressors

Model	MAE	R ²	RMSLE
CatBoost regressor	2.0008	0.9110	0.2021
Extreme gradient boosting	2.0140	0.9059	0.2005
Light gradient boosting machine	2.6135	0.8482	0.2415
AIRSTACKNET_PM25_rf	0.7028	0.9675	0.1263
AIRSTACKNET_PM25_et	.7552	0.9687	0.1243
AIRSTACKNET_PM25_ET_tuned	0.4720	0.9825	0.0902

As shown in [Tab. 4](#), We can note that Extra Trees Regressor, Random Forest Regressor, CatBoost Regressor, Extreme Gradient Boosting, and Light Gradient Boosting Machine perform the best with corresponding R² scores of 0.9679, 0.9490, 0.9110, 0.9059, and 0.8482, while Linear Regression, Huber Regressor, and Bayesian Ridge perform the worst in terms of R² corresponding scores of 0.1813,

0.0904, and 0.0014. Meanwhile, in terms of training time, we note that the Random forest regressor requires a lot of time with 351.2130 s, compared to the Linear regressor and Ridge regressor which needs only 0.1750 s. The findings from [Tabs. 5 and 6](#) are similar to those from [Tab. 4](#).

The computation time for each model during the training process is provided in the last column of [Tab. 4](#). Computation time is affected by the complexity of the selected base learners and the selected training data. Since Extra Trees Regressor and Random Forest Regressor, as bagging regressors, provide the best results in terms of predicting $PM_{2.5}$, O_3 and NO_2 values, meanwhile they require much time for training, we select these estimators as the meta-regressors for our AirStackNet model.

Moreover, we select CatBoost regressor, Extreme Gradient Boosting, and Light Gradient Boosting Regressor as base regressors in our AirStackNet Model since they demonstrate high R^2 scores and low mean errors. These results confirm the boosting and bagging models efficiency. Thus, our AirStackNet model will be composed of boosting models in the first layer for the base regressors, and of bagging models in the second layer for the meta-regressor. Obviously, the prediction performance of the proposed stacking model depends largely on the prediction performance of the base models. [Fig. 3](#) illustrates the comparison of the standard deviations of error metrics for AirStackNet and individual models. The relative stability of the proposed technique, with the minimum Standard Deviation of 0,0024 for R^2 , 0,0057 for RMSLE, 0,0029 for MAPE, with respect to its ingredients, demonstrates the efficiency of the model in handling stochastic variations. Indeed, In AirStackNet, we combine different training mechanisms and demonstrate their complementary properties.

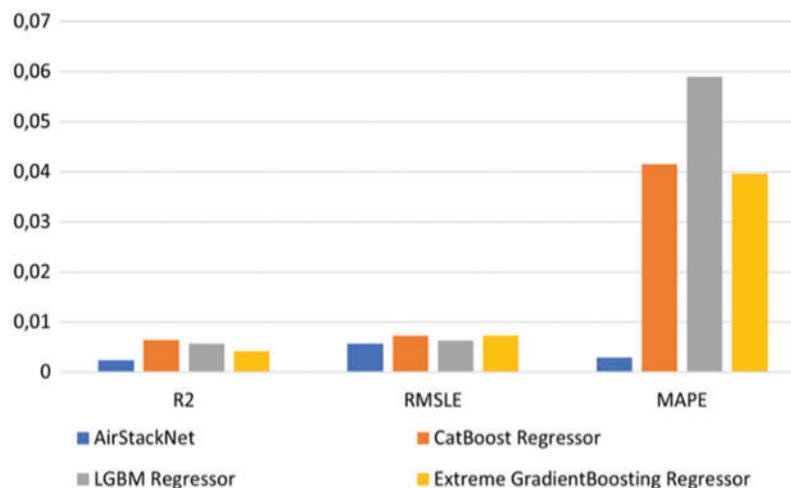


Figure 3: Comparison of the standard deviations of error metrics for AirStackNet and individual models

As shown in [Tab. 4](#), we note that Extra Trees Regressor, Random Forest Regressor, CatBoost Regressor, Extreme Gradient Boosting, and Light Gradient Boosting Machine perform the best with corresponding R^2 scores of 0.9679, 0.9490, 0.9110, 0.9059, and 0.8482, while Linear Regression, Huber Regressor, and Bayesian Ridge perform the worst in terms of R^2 corresponding scores of 0.1813, 0.0904, and 0.0014. meanwhile, in terms of training time, we note that the Random forest regressor requires a lot of time with 351.2130 s, compared to the Linear regressor and Ridge regressor which needs only 0.175 s. Since Extra Trees Regressor and Random Forest Regressor, as bagging regressors, provide the best results in terms of predicting $PM_{2.5}$ values, meanwhile they require much time for training, we select these estimators as the meta-regressors for our AirStackNet model. Moreover, we

select CatBoost regressor, Extreme Gradient Boosting, and Light Gradient Boosting Regressor as base regressors in our AirStackNet Model since they demonstrate high R^2 scores and low mean errors. These results confirm the boosting and bagging models efficiency. Thus, our AirStackNet model will be composed of boosting models in the first layer for the base regressors, and of bagging models in the second layer for the meta-regressor. Obviously, the prediction performance of the proposed stacking model depends largely on the prediction performance of the base models.

5.2 Second Case Study: Impact of Hyper-Parameters Tuning on Regression Performance

Based on the outcomes of the first case study that aims at selecting the estimators that build our AirStackNet model, we focus now on optimizing our stacking model by tuning the hyper-parameters of each estimator. For this reason, we perform k-10-fold cross-validation using grid search optimization. The obtained results are shown in Figs. 4 and 5. Fig. 4 displays the comparison of base regressor performance before and after tuning hyperparameters in terms of MAE, R^2 , and RMSLE metrics. As a result of tuning, MAE and RMSLE values have been decreased for the three regressors, while R^2 values have been increased for all as well. According to this finding, the base regressors' performances have been improved after tuning. The stability of the proposed AirStackNet model is mostly determined by the reliability of its ingredients, which are the base regressors. We can make inferences about the reliability of base regressors by looking at the standard deviation of each model between folds. Fig. 5 illustrates the variation of standard deviations between folds for the metrics (R^2 , RMSLE, MAPE) before and after regressors tuning. We note that the standard deviations of GradientBoosting Regressor, LightGBM Regressor, CatBoost Regressor have been decreased after tuning for the selected metrics, reflecting that the models produce stable results.

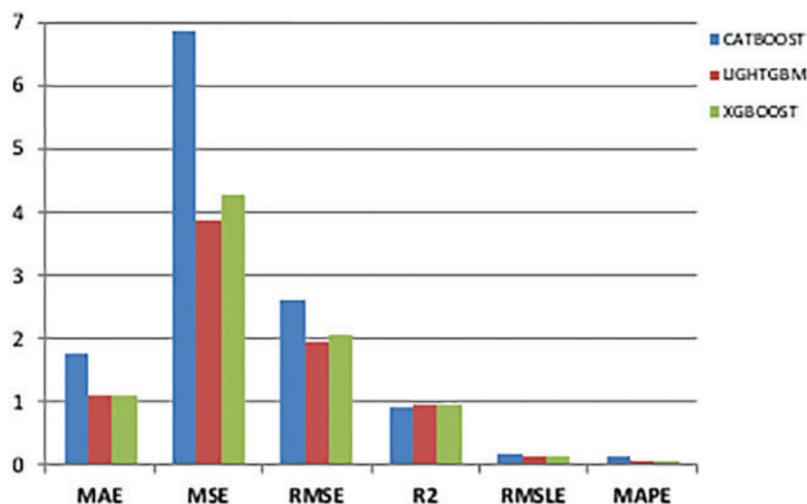


Figure 4: Hyper-parameters tuning using k-10-fold CV based on Mean

In this case study, AirStackNet model demonstrates that all of the individual models are combined to overcome the limitations of the low-performance prediction of single models. Therefore, it can be concluded that the proposed technique is well suited to the prediction of air pollution.

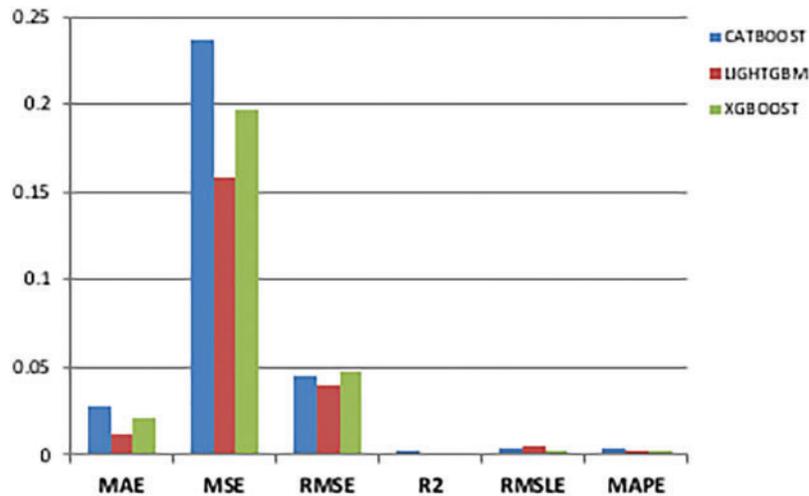


Figure 5: Hyper-parameters tuning using k-10-fold CV based on SD

5.3 Third Case Study: Assessment of the AirStackNet Model

After tuning and evaluating the base estimators of the stacking model, we focused on assessing the performance of our AirStackNet *vs.* its base regressors which are LightGBM, CatBoost, and XGBoost, when predicting $PM_{2.5}$ values. In this experiment, AirStackNet has Extra Trees regressor as the meta-regressor.

The obtained results are illustrated in [Tab. 7](#). By using AirStackNet, the MAE values were dropped to 0.7552 compared to 2.008 with CatBoost Regressor, 2.0140 with Extreme Gradient Boosting Regressor, and 2.6135 with LightGBM Regressor. Similar finding was found for RMSLE metric. As for R^2 , AirStackNet outperforms the individual regressors with 0.9687 *vs.* 0.9110 for CatBoost Regressor, 0.9059 for Extreme Gradient Boosting Regressor and 0.8482 for LightGBM Regressor. Based on these results, we can conclude that the $PM_{2.5}$ prediction was improved with AirStackNet. This improvement is due to the use of different boosting algorithms having their own specific rules that can greatly reflect the advantages of this diversity. Indeed, this diversity makes the stacking result more effective and accurate which makes the predicted effect more justified. Furthermore, when comparing the two last rows in the [Tab. 7](#), hyperparameter optimization for individual regressors has improved prediction performance of AirStackNet by minimizing MAE to 0.4720 and RMSLE to 0.0902, reflecting the importance of the regressors tuning step.

Table 7: Comparison of AirStackNet variants performance *vs.* individual base regressors O_3

Model	MAE	R^2	RMSLE
AIRSTACKNET_NO2_et	2.0380	0.9596	0.1593
AIRSTACKNET_NO2_Tuned	1.2687	0.9782	0.1142
AIRSTACKNET_O3_et	0.8060	0.9619	0.2954
AIRSTACKNET_O3_Tuned	0.4445	0.9822	0.1925

To validate the generalization of our proposed stacking model, we conducted experiments over the O_3 and NO_2 pollutants values. the corresponding results are displayed in [Tab. 7](#). It can be seen from

this table the improvement of performance when using tuned base regressors. In addition, we can note that the range of performance is similar for all pollutants, which reflects the stability and reliability of our proposed model for all air pollutants prediction.

As shown in [Tab. 6](#), the prediction performance is improved using AirStackNet variants compared to individual models. AirStackNet_PM25_RF achieves a low MAE value of 0.7028 while CatBoost, XGBoost, and LightGBM achieve MAE values superior to 2. Thus, AirStackNet_PM25_RF outperforms these base regressors. Similarly, AirStackNet_PM25_ET achieves a low MAE value of 0.7552. This improvement is due to the use of different boosting algorithms having their own specific rules that can greatly reflect the advantages of this diversity. Indeed, this diversity makes the stacking result more effective and accurate which makes the predicted effect more justified.

To further study the impact of the stacking model, we compare the two variants in terms of MAE, R^2 , and RMSLE. The obtained results illustrate that AirStackNet_PM25_RF and AirStackNet_PM25_ET achieve similar results with a slight improvement of AirStackNet_PM25_ET in terms of R^2 , and as light improvement of AirStackNet_PM25_RF in terms of MAE. This similarity is justified by the fact that Random Forest and Extra Trees are two ensemble methods having a lot of common rules. Both are composed of a large number of decision trees and have the same growing tree procedure.

The last variant AirStackNet_PM25_ET_tuned consists of stacking tuned base regressors and using Extra Trees as meta regressor. The performance results of this variant report an improvement in the prediction by achieving the highest R^2 score with 0.9825 and the lowest error rate in terms of MAE and RMSLE with corresponding scores 0.4720 and 0.0902. This improvement is due to the hyperparameters tuning of the base regressors.

To validate our proposed stacking model, we conducted experiments over the O_3 and NO_2 pollutants values. the corresponding results are displayed in [Tab. 7](#). It can be seen from this table the improvement of performance when using tuned base regressors. In addition, we can note that the range of performance is similar for all pollutants, which reflects the stability and reliability of our proposed model.

5.4 Forth Case Study: Optimization of the AirStackNet Structure

Constructing an optimal ensemble of models is challenging, as it entails selecting which models should be included. A successful ensemble depends heavily on the quality and variety of its individual learners. These learners need to be accurate and diverse enough in order to reflect the structure of the data. In particular, bagging and boosting represent different aspects of the data, and when combined, can reveal the entire data space more effectively.

Based on this assumption, and in order optimize the structure of AirStackNet, four variants of the AirStackNet model were derived, varying in the way the base regressors and meta regressors were combined. In fact, we would like to check the influence of base regressor nature as well as the meta regressor on the prediction results in our stacking model. The obtained results are reported in [Tab. 6](#). As such, in the two first rows of [Tab. 6](#), we used the Random Forest as a meta regressor in the first variant AirStackNet_PM25_RF, while we used the Extra Trees regressor in the second variant AirStackNet_PM25_ET. We compare the two variants in terms of MAE, R^2 , and RMSLE. The obtained results illustrate that AirStackNet_PM25_RF and AirStackNet_PM25_ET achieve similar results with a slight improvement of AirStackNet_PM25_ET in terms of R^2 , and as light improvement of AirStackNet_PM25_RF in terms of MAE. This similarity is justified by the fact that Random Forest and Extra Trees are two ensemble methods having a lot of common rules. Both

are composed of a large number of decision trees and have the same growing tree procedure. By changing the bagging meta-regressor by a linear meta-regressor such as Ridge and Lasso as illustrated for AirStackNet_PM25_Lasso and AirStackNet_PM25_Ridge, In terms of reported errors metrics values, we can observe a degradation of performance. It is primarily due to the probability of linear regression being unbiased and producing smaller residuals. This variant Stacking (ET, RF, LightGBM) is an ensemble model using the bagging models as base learners and the boosting algorithms as meta-learners. Our aim is to check the capability of bagging to perform the role of base regressors. Unfortunately, This ensemble model cannot outperform the two first variants. Indeed, the boosting learners aim to provide various fitted prediction, while the role of Bagging is to combine all the base boosting learners together in order to “smooth out” their predictions. As such, the optimal structure of AirStackNet is to use boosting for base learners and Bagging for meta-learners. Aside from these variants, one state-of-the-art benchmark is used to compare with it the results of the designed model AirStackNet. This benchmark [16] does not perform well comparing to our proposed AirstackNet model due to the use of the linear regression in the meta-learner stage. Fig. 6 shows some example smartphone photographs from the test dataset with corresponding ground-truth $PM_{2.5}$ concentrations and the $PM_{2.5}$ concentrations estimated by the AirStackNet model. It can be seen that the estimated $PM_{2.5}$ concentrations correlate well with the $PM_{2.5}$ ground-truth concentrations.

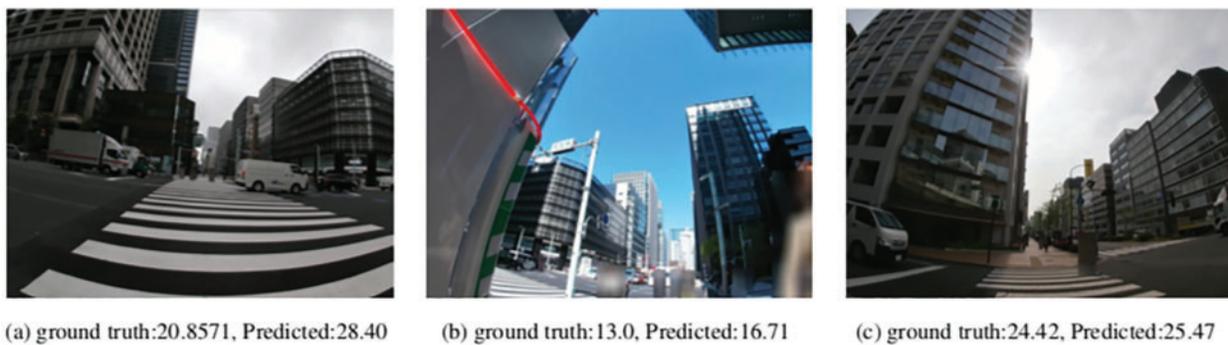


Figure 6: Some example smartphone photographs from the test dataset with corresponding ground-truth $PM_{2.5}$ concentrations and the $PM_{2.5}$ concentrations estimated by the AirStackNet model

6 Conclusion

This research contributed to solving the challenge raised in MediaEval2020’s Insights for Wellbeing task “Can personal air quality be predicted using images captured by personal devices and some open source data?”. Our three-step method predicts local AQI values by imputing missing data using the ball tree algorithm, extracting haze-relevant features, and building the AirStackNet model. The model relies primarily on the stacking ensemble learning approach in order to create an efficient, scalable, and reliable air quality prediction. We mainly contributed to justifying the stacked structure of the model. With our realized experiments, we showed that boosting regressors are highly efficient as base learners, whereas bagging regressors perform well as meta-learners in the stacking model structure. To refine the model, we proved the importance of hyperparameter-tuning the base learners using k-10 cross-validation. The obtained small error variance values confirm the reliability of the base learners and the stability of our model. Comparing AirStackNet to its base regressors and to existing methods, the results showed AirStackNet performed the best which reflects its effectiveness

in terms of used error metrics. Thus, the proposed stacking model structure can serve as a useful and generalized reference model for air quality prediction.

To make our model even more effective, we plan to add pre-training models like VGGNet, ResNet, Inception, and Xception, as well as new semantic features about outdoor areas, traffic, and buildings. Additionally, we plan to integrate our proposed model into an API and create a mobile application that will let citizens take a picture of the sky and get the air quality index by calling the API encapsulating our proposed model. By using this application, users will be able to identify the predominant pollutants at a place and the risks they face.

Acknowledgement: The authors extend their appreciation to the Deputyship for Research and Innovation, Ministry of Education in Saudi Arabia for funding this research through project number PNU-DRI-RI-20-033.

Funding Statement: The authors extend their appreciation to the Deputyship for Research and Innovation, Ministry of Education in Saudi Arabia for funding this research through project number PNU-DRI-RI-20-033.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] Y. Hu, M. Yao, Y. Liu and B. Zhao, "Personal exposure to ambient PM_{2.5}, PM₁₀, O₃, NO₂, and SO₂ for different populations in 31 Chinese provinces," *Environment International*, vol. 144, no. 1, pp. 106018, 2020.
- [2] B. Karimi, B. Shokrinezhad and S. Samadi, "Mortality and hospitalizations due to cardiovascular and respiratory diseases associated with air pollution in Iran: A systematic review and meta-analysis," *Atmospheric Environment*, vol. 198, no. 1, pp. 438–447, 2019.
- [3] D. H. Nguyen, T. L. Nguyen-Tai, M. T. Nguyen, T. B. Nguyen and M. -S. Dao, "MNR-Air: An economic and dynamic crowdsourcing mechanism to collect personal lifelog and surrounding environment dataset. A case study in Ho Chi minh city, Vietnam," *Proceedings of MultiMedia Modeling (MMM 2021)*, vol. 12573, pp. 206–217, 2021.
- [4] Y. Zheng, F. Liu and H. -P. Hsieh, "U-Air: When urban air quality inference meets big data," in *Proc. of the 19th SIGKDD Conf. on Knowledge Discovery and Data Mining (KDD 2013)*, New York, NY, USA, pp. 1436–1444, 2013.
- [5] Q. Zhang, F. Fu and R. Tian, "A deep learning and image-based model for air quality estimation," *Science of the Total Environment*, vol. 724, no. 1, pp. 138178, 2020.
- [6] T. La, M. Dao, K. Tejima, R. U. Kiran and K. Zettsu, "Improving the awareness of sustainable smart cities by analyzing lifelog images and iot air pollution data," in *Proc. of 2021 IEEE Int. Conf. on Big Data (Big Data)*, Orlando, FL, USA, IEEE, pp. 3589–3594, 2021.
- [7] A. Ksibi, A. Salhi, A. S. D. Alluhaidan and S. A. El-Rahman, "Insights for wellbeing: Predicting personal air quality index using regression approach," in *Proc. of MediaEval, Delft University of Technology*, Netherlands, 2020.
- [8] M. -S. Dao, P. Zhao, T. Sato, K. Zettsu, D. -T. Dang-Nguyen *et al.*, "Overview of mediaeval 2020 insights for wellbeing: Multi-modal personal health lifelog data analysis," in *Proc. of MediaEval, Delft University of Technology*, Netherlands, 2019.
- [9] M. Dao, K. Zettsu and R. U. Kiran, "IMAGE-2-AQI: Aware of the surrounding air qualification by a few images," in *Proc. of 34th Int. Conf. on Industrial, Engineering and other Applications of Applied Intelligent Systems, IEA/AIE 2021*, Kuala Lumpur, Malaysia, July 26–29, 2021, pp. 335–346, 2021.

- [10] P. B. Vo, T. D. Phan, M. -S. Dao and K. Zettsu, "Association model between visual feature and AQI rank using lifelog data," in *Proc. of 2019 IEEE Int. Conf. on Big Data (Big Data)*, Los Angeles, CA, USA, pp. 4197–4200, 2019.
- [11] Z. Wang, W. Zheng, C. Song, Z. Zhang, J. Lian *et al.*, "Air quality measurement based on double-channel convolutional neural network ensemble learning," *IEEE Access*, vol. 7, pp. 145067–145081, 2019.
- [12] D. Q. Duong, Q. M. Le and D. T. Nguyen, "A2QI: An approach for air pollution estimation in mediaeval 2020," in *Proc. of the MediaEval 2020 Workshop*, Online, vol. 2882, 2020.
- [13] B. Zhai and J. Chen, "Development of a stacked ensemble model for forecasting and analyzing daily average PM_{2.5} concentrations in Beijing, China," *Science of the Total Environment*, vol. 635, pp. 644–658, 2018.
- [14] C. Y. Lin, Y. S. Chang and S. Abimannan, "Ensemble multifeatured deep learning models for air quality forecasting," *Atmospheric Pollution Research*, vol. 12, no. 5, pp. 101045, 2021.
- [15] D. Q. Duong, Q. M. Le, T. Nguyen-Tai, D. Bo, D. Nguyen *et al.*, "Multi-source machine learning for AQI estimation," in *Proc. of 2020 IEEE Int. Conf. on Big Data (IEEE BigData 2020)*, Atlanta, GA, USA, pp. 4567–4576, 2020.
- [16] D. Q. Duong, Q. M. Le, T. Nguyen-Tai, H. D. Nguyen, M. Dao *et al.*, "An effective AQI estimation using sensor data and stacking mechanism," in *Proc. of the 20th Int. Conf. on New Trends in Intelligent Software Methodologies, Tools and Techniques, SoMeT 202*, Cancun, Mexico, vol. 337, pp. 405–418, 2021.
- [17] P. Zhao and K. Zettsu, "Decoder transfer learning for predicting personal exposure to air pollution," in *Proc. of 2019 IEEE Int. Conf. on Big Data (Big Data)*, Los Angeles, CA, USA, pp. 5620–5629, 2019.
- [18] Y. Xu, W. Yang and J. Wang, "Air quality early-warning system for cities in China," *Atmospheric Environment*, vol. 148, pp. 239–257, 2017.
- [19] H. Fujita and H. Perez-Meana, "New trends in intelligent software methodologies," in *Proc. of the 20th International Conf. on New Trends in Intelligent Software Methodologies, Tools and Techniques*, Cancun, Quintana Roo, Mexico, IOS Press, 2021.
- [20] Y. -C. Liang, Y. Maimury, A. H. -L. Chen and J. R. C. Juarez, "Machine learning-based prediction of air quality," *Applied Sciences*, vol. 10, no. 24, pp. 9151, 2020.
- [21] Y. Liu, J. Nie, X. Li, S. H. Ahmed, W. Y. B. Lim *et al.*, "Federated learning in the sky: Aerial-ground air quality sensing framework with UAV swarms," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9827–9837, 2021.
- [22] K. He, J. Sun and X. Tang, "Single image haze removal using dark channel prior," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Miami, FL, USA, pp. 1956–1963, 2009.
- [23] D. Q. Duong, Q. M. Le and D. T. Nguyen, "A2QI: An approach for air pollution estimation in mediaeval 2020," in *Proc. of the MediaEval 2020 Workshop*, Online, vol. 2882, 2020.
- [24] T. -Q. Nguyen, D. -H. Nguyen and L. T. T. Nguyen, "Personal air quality index prediction using inverse distance weighting method," in *Proc. of the MediaEval 2020 Workshop*, Online, vol. 2882, 2020.
- [25] P. Zhao and K. Zettsu, "Convolution recurrent neural networks for short-term prediction of atmospheric sensing data," in *Proc. of IEEE Int. Conf. on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, Halifax, NS, Canada, pp. 815–821, 2018.
- [26] H. Song, K. J. Lane, H. Kim, H. Kim, G. Byun *et al.*, "Association between urban greenness and depressive symptoms: Evaluation of greenness using various indicators," *International Journal of Environmental Research and Public Health*, vol. 16, no. 2, pp. 173, 2019.
- [27] N. -T. Nguyen, M. -S. Dao and K. Zettsu, "Complex event analysis for traffic risk prediction based on 3D-CNN with multi-sources urban sensing data," in *Proc. of IEEE Int. Conf. on Big Data (Big Data)*, Los Angeles, CA, USA, pp. 1669–1674, 2019.
- [28] M. S. Vahdatpour, H. Sajedi and F. Ramezani, "Air pollution forecasting from sky images with shallow and deep classifiers," *Earth Science Informatics*, vol. 11, pp. 413–422, 2018.
- [29] M. Chalela, E. Sillero, L. Pereyra, M. A. Garcia, J. Cabral *et al.*, "GriSPy: A python package for fixed-radius nearest neighbors search," *Astronomy and Computing*, vol. 34, pp. 100443, 2021.

- [30] S. G. Narasimhan and S. K. Nayar, "Vision and the atmosphere," *International Journal of Computer Vision*, vol. 48, pp. 233–254, 2004.
- [31] Z. Chen, T. Zhang, Z. Chen, Y. Xiang, Q. Xuan *et al.*, "A high-resolution vision-based air quality dataset," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–10, 2021.
- [32] R. Liu, Z. Li and J. Jia, "Image partial blur detection and classification," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Anchorage, AK, USA, pp. 1–8, 2008.
- [33] Z. Zhang, H. Ma, H. Fu, L. Liu and C. Zhang, "Outdoor air quality level inference via surveillance cameras," *Mobile Information Systems*, vol. 2016, pp. 1–10, 2016.
- [34] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992.