Forecasting Fine-Grained Air Quality Based on Big Data

Yu Zheng[1,2], Xiuwen Yi[2,1], Ming Li[1], Ruiyuan Li[1], Zhangqing Shan[3,1], Eric Chang[1], Tianrui Li[2]
[1]Microsoft Research, Beijing, China
[2]Southwest Jiaotong University, Chengdu, Sichuan, China
[3]Fudan University, Shanghai, China
{yuzheng, v-xiuyi, mingl, v-ruiyli, v-zhasha, echang}@microsoft.com, trli@swjtu.edu.cn

ABSTRACT
In this paper, we forecast the reading of an air quality monitoring station over the next 48 hours, using a data-driven method that considers current meteorological data, weather forecasts, and air quality data of the station and that of other stations within a few hundred kilometers. Our predictive model is comprised of four major components: 1) a linear regression-based temporal predictor to model the local factors of air quality, 2) a neural network-based spatial predictor to model global factors, 3) a dynamic aggregator combining the predictions of the spatial and temporal predictors according to meteorological data, and 4) an inflection predictor to capture sudden changes in air quality. We evaluate our model with data from 43 cities in China, surpassing the results of multiple baseline methods. We have deployed a system with the Chinese Ministry of Environmental Protection, providing 48-hour fine-grained air quality forecasts for four major Chinese cities every hour. The forecast function is also enabled on Microsoft Bing Map and MS cloud platform Azure. Our technology is general and can be applied globally for other cities.

Categories and Subject Descriptors
H.2.8 [**Database Management**]: Database Applications - *data mining, Spatial databases and GIS*;

Keywords
Urban computing; urban air; air quality forecast; big data.

1. INTRODUCTION
People are increasingly concerned with air pollution, which impacts human health and sustainable development around the world. Many cities have built air quality monitoring stations to inform people about urban air quality, e.g. the concentration of PM2.5 (particulate matter) and PM10, every hour. Besides monitoring, there is a rising demand for the prediction of future air quality, which can inform people's decision making (e.g. whether to go for picnic or jogging in a park) and governments' policy making (such as issuing pollution alerts or performing a pollution control).

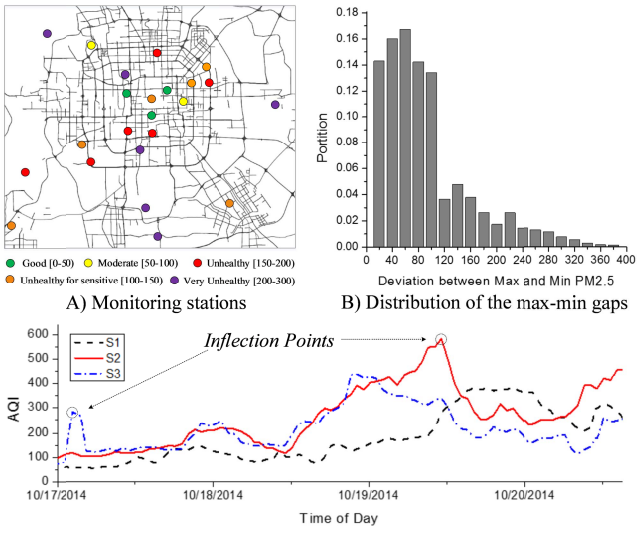
Predicting urban air quality, however, is very challenging for the following three reasons: *First*, while urban air quality is affected by multiple complex factors, such as traffic flow, meteorology, and land use [11][14], we do not have sufficient and accurate data to model each factor. For example, it is almost impossible to obtain the accurate pollution emissions data of every vehicle and factory in a real time manner. Likewise, weather forecasts have not been able to tell us exactly when a wind will blow and how long it will last for.

KDD '15, August 11 - 14, 2015, Sydney, NSW, Australia

DOI: http://dx.doi.org/10.1145/2783258.2788573

Second, urban air changes over location and time significantly because of these complex factors. As shown in Figure 1 A), there are 22 air quality monitoring stations in Beijing's urban areas. If we calculate the gap between the maximum and minimum AQIs (Air Quality Indexes) of PM2.5 from these stations at the same hour, as illustrated in Figure 1 B), about 40 percent of time slots have a gap larger than 100, which denotes a two-level difference in pollution (i.e. when the air quality of a location is moderate, another one is unhealthy). Moreover, as depicted in Figure 1 C), the AQIs of three stations change over time very differently. For instance, while the readings of S_1 and S_2 increasing in early part of Oct. 19, 2014, that of S_3 was decreasing. So, we need to predict the air quality of different stations (or even the different time slots of the same station) by using different models. In other words, a general prediction of the overall air quality in a city is not useful enough to inform people's decision making.

Third, there are some inflection points where air quality changes very sharply. This may be caused by unusual weather conditions, such as rain storms or strong winds. As such inflection instances are very rare in observation, a general statistic model will be dominated by normal instances, and therefore cannot predict such inflections or sudden changes very well.



A) Monitoring stations
B) Distribution of the max-min gaps
C) AQI of different stations changing over time of day

Figure 1. Deviation between different monitoring stations' PM2.5: A circle shown in A) denotes a station, and its color means the level of air pollution, as described in the bottom of the figure. The air quality of these stations were dramatically different at 10am Mar. 13, 2014.

To address the aforementioned issues, we predict the air quality over the next 48 hours for each monitoring station. As shown in Figure 2, in the first 6 hours, we predict a real-valued AQI for each kind of air pollutant, at each hour in each station. For the next 7-12, 12-24, and 24-48 hours, we predict a max-min range of AQIs at the corresponding time interval. That is, a coarser granularity of forecast is provided for a farther future. We continuously predict these values every hour for each station.

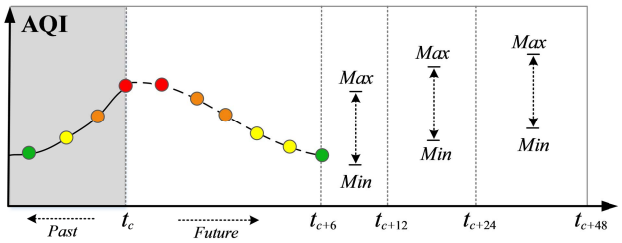


Figure 2: Format of the air quality forecasts

In our method, we train a hybrid predictive model for each kind of pollutant at each station and in different time slots, based on the following three types of data over a period of time: 1) The air quality at the current time and over the past few hours; 2) The meteorological data (such as humidity, temperature, and wind speed) at the current time and over the past few hours; 3) Weather forecasts at the future time we are going to predict. More specifically, for the first two datasets, we need the data of the station we are going to predict and those of other stations within a circle distance (of a few hundred kilometers) to the station. For the first six hours, we train a model for each hour at each station. With respect to the following three ranges, we train a model to predict the maximum and minimum AQIs respectively. Our contribution has four parts:

- We propose a multi-view-based hybrid model that predicts future air quality with inaccurate and insufficient data. The model handles the spatial correlation of air quality among different locations and the temporal dependency of air quality at a location, using non-overlapped features and different machine learning models. It then combines the spatial and temporal predictions dynamically according to weather conditions.
- The inflection predictor in our hybrid model significantly improves the capability of predicting sudden changes of air quality caused by extreme weather conditions.
- We evaluate our model with data from 43 cities in China, achieving a precision greater than 0.75 in the first six hours. Our method significantly outperforms baselines when dealing with general instances, and has a 1.5 times higher accuracy when handling sudden drops.
- The system has been deployed, using a framework that combines the cloud with clients. The cloud is located at Microsoft Azure, continuously collecting real-time data and forecasting air quality. People can access the fine-grained air quality information by using either a mobile app, called Urban Air [2], or through a public website [1]. The technology has also been shipped into Bing Maps China [3]. Additionally, we have deployed the system in Chinese Ministry of Environmental Protection, providing fine-grained air quality for the current time and future hours to inform governments' decision making. The datasets have been released in [17].

The rest of the paper is organized as follows: In Section 2, we present an overview of our system. Section 3 details the predictive model. We evaluate our method in Section 4. Section 5 summarizes related work. We draw conclusions in Section 6.

2. OVERVIEW

2.1 System Architecture

Figure 3 presents the architecture of our system, which consists of three parts: External Data Sources, Cloud and Clients. The Data Sources include a list of public websites and public/private web services providing real-time meteorological data, weather forecasts, and air quality data of different cities. The Cloud is based on Microsoft Azure, hosting five major components of our system. The Data Collector continuously collects real-time data from external data sources, through web service interfaces or by crawling web pages. The collected data is stored in a cloud database. For various reasons, a few air quality monitoring stations occasionally may not have readings; same with the meteorological data. Thus, the Data Supplement component tries to fill the missing values in the collected data based on their spatial or temporal neighbors. The Predictive Model components is comprised of a collection of models, each of which predicts the air quality for a station and at a time interval. The prediction results are then stored in the cloud database for the access of the Web Service component, which provides interfaces to two types of clients: mobile apps and websites. To have a stable and robust system, we set a monitor to continuously check the availability of data sources and the performance of web services as well as the status of other components.

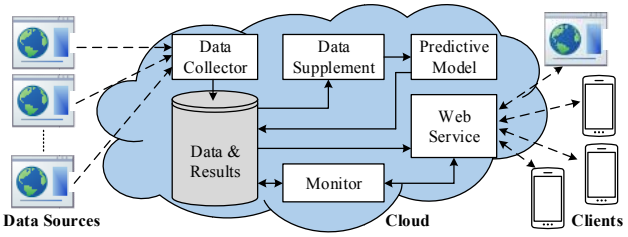


Figure 3. Architecture of our system

2.2 User Interfaces

Figure 4 presents the website of Urban Air [1], where an icon on the map stands for a monitoring station and the number associated with an icon denotes its AQI; the smaller the number is, the better the air quality is. The color of an icon is determined in accordance with its air quality, e.g. "green" means a "good" and "yellow" denotes "moderate" by Chinese AQI standards. After clicking the most right (trend) tab on the floating tool bar, we will see a time line, with which a user can check air quality forecasts of a specific future time interval. Users can also check the future air quality of all stations changing over time by clicking the start button on the left terminal of the time line. By clicking a specific station on the map, users will see a pop-up chart showing a curve of air quality forecast. The number on the top of each time segment is an accuracy of the prediction measured by the data at the station in the past 48 hours. The accuracy of a maxi-min range is measured by its mean value against the mean value of the true range.

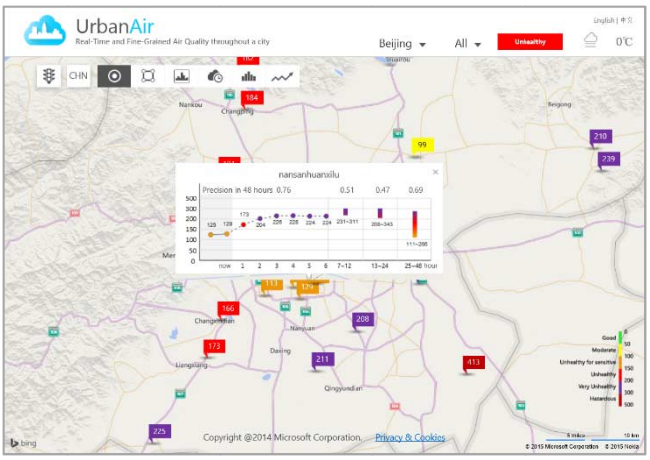
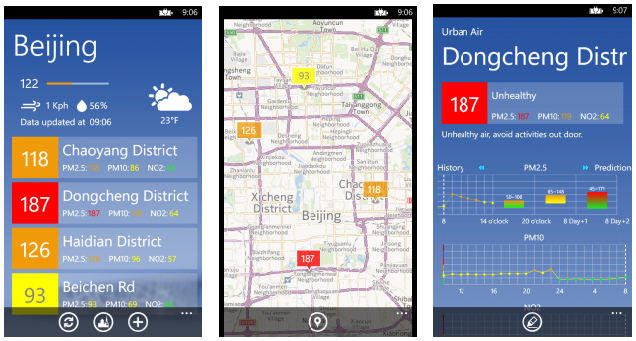


Figure 4. Web user interface of Urban Air

Figure 5 presents the user interface of mobile clients. As depicted in Figure 5 A), a user has selected four locations, such as home and work places, to monitor on their mobile phone. Here, each banner represents one location and the number shown in each banner is the AQI of the location. Each location was selected by pressing and holding the corresponding venue on a map, as shown in Figure 5B), where an icon stands for a venue that a user has selected. Our mobile client will automatically name a selected venue according to the titles of POIs and road networks around the venue. Users can then modify the name to some semantic title, such as home. By clicking a banner in the location list, users can see not only the historical air quality of a location in the past 24 hours but also the forecast of next 48 hours, as illustrated in Figure 5 C).



A) List view B) Map view C) Chart view

Figure 5. Mobile interface of Urban Air

2.3 Framework of the Predictive Model

Figure 6 presents the framework for the predictive model, consisting of four components: a (local) temporal predictor, a (global) spatial predictor, an inflection predictor, and a prediction aggregator.

The *Temporal Predictor* predicts the air quality of a station in terms of the data about the station, such as local meteorology, AQIs from the past few hours and the weather forecast for where the station is located. Alternatively, we can say the temporal predictor predicts air quality using *local* data, considering the prediction more from its own historical and future conditions. Specifically, the temporal predictor is based on a linear regression (*LR*), which models the local air quality regression process.

Instead, the *Spatial Predictor* considers spatial neighbor data, such as the AQIs and the wind speed at other stations, to predict a station's future air quality. Intrinsically, the air quality of different locations has a spatial correlation as pollutants are dispersed from one place to another. The *Spatial Predictor* is based on an artificial neural network (*ANN*), modeling the spatial correlation and predicting air quality from other locations' points of view.

The two predictors generate their own predictions independently for a station, which are combined by the *Prediction Aggregator* dynamically according to the current weather conditions of the station. Sometimes, local prediction is more important, while spatial prediction should be given a higher weight on other occasions (e.g. when a wind blows strongly). As the deviation between two consecutive hours' AQIs (ΔAQI) is usually smaller than the AQI itself, the two predictors predict the deviation rather than the original AQI.

There are three reasons we need to devise three separate (spatial, temporal, and aggregator) predictors rather than a single predictor: 1) *From the feature space's perspective*, the features used by the spatial and temporal predictors do not have any overlap, providing different views on a station's air quality. 2) *From the model's perspective*, the spatial and temporal predictors model local factors and global factors respectively, which have significantly different properties. For example, the local is more about a regression problem, while the global is more about a non-linear interpolation. Thus, they should be handled with different techniques. 3) *From the parameter learning's perspective*, feeding all features into a single model results in a big model with many parameters to learn. However, the training data is limited. For instance, we only have one year of AQI data for a city. Decomposing a big model into three organically coupled small models scales down the parameter spaces tremendously, leading to more accurate learning and therefore prediction.

In some cases, e.g. when strong winds or rain storms come, the air quality of a location drops tremendously in a short time period. Such kind of sharp drops are hard to predict, as their presence is very small in the entire observation. To address this issue, we pick out such sudden drop instances to train a separate *Inflection Predictor*. We also learn some conditions that significantly differentiate these drop instances from normal cases, e.g. when the wind speed is higher than a threshold. Once one of these conditions holds, the *Inflection Predictor* will be invoked to generate a ΔAQI, which will be appended to the original AQI with the output of the *Prediction Aggregator* to calculate the final prediction.

As different stations are located in different environments and different air pollutants vary by location and time, we build such a hybrid model for each kind of air pollutant, at each station and for different time intervals. More specifically, we train a model respectively for each hour in the next six hours, and two models for each time interval (from 7 to 48 hours) to predict its maximum and minimum values. All these models are re-trained every a few months in an offline process and generate an online prediction every hour.

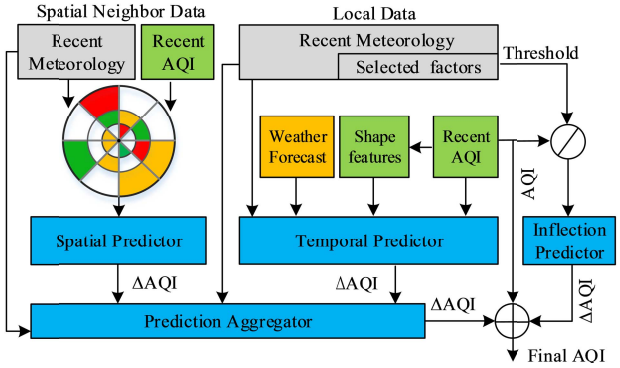


Figure 6. Framework of the predictive model

3. Hybrid Predictive Model

3.1 Temporal Predictor

The temporal predictor models the trend of air quality of a station based on four types of data: 1) the AQIs of the past h hours at the station; 2) the local meteorology (such as sunny/overcast/cloudy/foggy, humidity, wind speed, and direction) at the current time t_c; 3) time of day and day of the week; 4) the weather forecasts (including sunny/overcast/cloudy, wind speed, and wind direction) of the time interval we are going to predict. These features have been proven relevant to air quality in past literature [4][11][14].

Intuitively, the current status has different degrees of impact to different future time intervals. Thus, as illustrated in Figure 7, we pair the inputs (shown in the broken rectangle) with the air quality of different time intervals ($t_{c+1}, t_{c+2}, \ldots, t_{c+48}$) to formulate different training sets, which are used to respectively train different models corresponding to different time intervals. Each blue broken arrow shown in Figure 7 denotes a temporal predictor. Over the

next six hours, we train a model for each hour. With respect to the next 7 to 48 hours, which are divided into three time intervals (7-12, 13-24, 25-48), we train two models to predict the maximum and minimum AQIs of each time interval respectively. This is the same for spatial predictors which will be detailed later. The first three parts of inputs are the same in different temporal predictors, while the only difference between different predictors' inputs lies in the weather forecast (i.e. the fourth part).

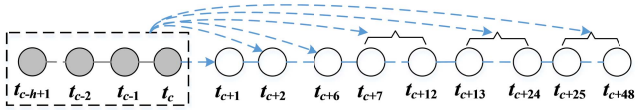


Figure 7. Illustration of the temporal predictor

A linear regression is employed to model the local change of air quality. The categorical features are converted into numeric values, e.g. using (0, 1) to denote (not sunny, sunny) respectively. We do not conduct an iterative moving prediction (e.g. using the prediction of t_{c+1} as an input to predict later hours) for two reasons. First, the new prediction will bring errors to later rounds of prediction. Second, a weather forecast is coarser and less accurate than current meteorological data. Some features (e.g. humidity) are even missing in the weather forecast of many cities. Though one can capture the general trends in air quality at a location using a regression process, the temporal predictor has its weaknesses, e.g. it cannot well handle sudden changes and pollution coming from other places.

3.2 Spatial Predictor

Beside local emissions, the air quality of a location also depends on its neighbors, as air pollutants are dispersed among different locations. For example, if there are pollution emissions from a factory that is 20 kilometers away from a station and the wind happens to blow them towards the station, the air quality of the station will be become bad soon after. To model spatial correlations in air quality at different locations, we devise a spatial predictor which predicts the air quality of a location based on other stations' status consisting of AQIs and meteorological data. The spatial neighbors of a station include not only nearby stations but also the stations located in adjacent cities. To model the impact from different locations, the distance between the station and its neighbors ranges from a few kilometers to several hundred kilometers. Although we do not have first-hand pollution emission data, the stations that have been built can be regarded as sensors sending signals to our spatial predictors.

As shown in Figure 8 A), to build a spatial predictor for a station s, we first partition the spatial space into regions by using three circles with different diameters. The outmost circle has the largest diameter (e.g. 300km), and the innermost one has the least (e.g. 30km). The three circles share a common center (i.e. station s denoted by the black point) and are further segmented by four lines pointing to different angles. We then project other stations onto the regions bound by the line fragments and circles, according to each station's geo-coordinates. To simplify the model, the stations falling outside the biggest circle are not considered in the spatial predictor. As illustrated in Figure 8 B), we aggregate the meteorological data and air quality readings from the stations that are located in the same region. When a region has more than one station, we calculate the average AQI for a given kind of air pollutant; same for temperature and humidity. The wind direction of a region is determined by the mode of the data. As a result, each region will only have one set of aggregated air quality readings and meteorology, which will be fed into the spatial predictor to predict the future air quality for s. We conduct the same process of spatial partition and aggregation for different stations.

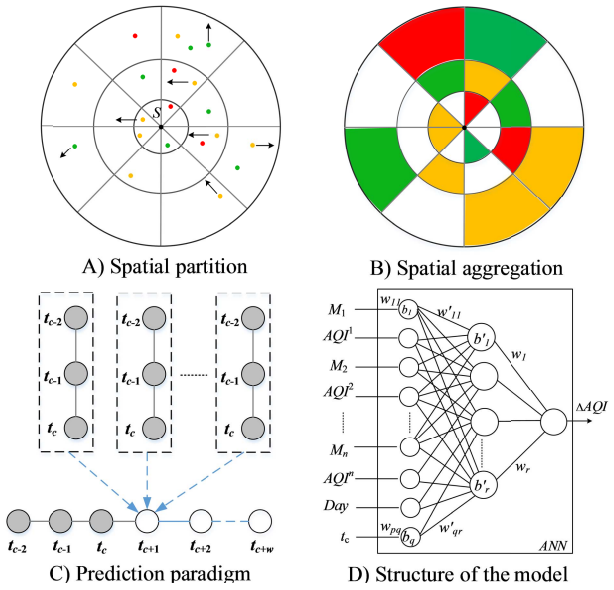


Figure 8. Illustration of the spatial predictor

The reason that we partition a spatial space into regions and then aggregate the readings from other stations are threefold:

1) If we directly feed all the data from a station's neighbors into a machine learning model, the number of parameters increases quickly in proportion to the number of stations. This causes a trouble for model training and therefore prediction. Remember that we do not have sufficient data to train a very big model. The more parameters involved in a model, the more training data we need to find a set of proper parameters. The partition and aggregation significantly reduces the number of inputs, enhancing training quality. It also sets an upper bound for the input (no matter how many new stations will be built in the future), as the number of regions is fixed given the spatial partition process.

2) The information from nearby stations is somehow redundant or even sometimes contradictory. For example, the wind directions of two nearby stations could be opposite, as wind may be affected by urban canyons. Without a proper aggregation, the spatial predictor will be confused by the chaotic input.

3) The partition and aggregation carry a semantic meaning, denoting different regions' impacts (to station s) varying by distance and angle. By setting different diameters for different circles, the partition provides a coarser granularity for a farther region and a finer granularity for closer regions. In other words, we aggregate the data of a larger area for the regions located in a more outward ring.

As demonstrated in Figure 8 C), after the spatial partition and aggregation, we formulate a time series for each region with at least one station. Other regions without stations are called empty regions, which are not considered in a spatial predictor. In the time series (denoted as a broken-bound box), each node stands for the aggregated information of a corresponding hour. For a non-empty region i, we extract the following features from its time series: the AQI of the past three hours ($\boldsymbol{AQI}^i$) and meteorological features (M^i), including the wind speed and direction, at the current time t_c. As depicted in Figure 8 D), $\boldsymbol{AQI}^i$ and M^i of the non-empty regions are fed into an artificial neural network whose output is the AQI of the station at the specific time interval we are going to predict. In the implementation, we predict the deviation between the AQIs of current time t_c and the future time interval t_{c+w}, i.e. $\Delta AQI = AQI_{t_c} - AQI_{t_{c+w}}$, because the distribution space of ΔAQI is much narrower than $AQI_{t_{c+w}}$. For example, Figure 9 A) presents the

distribution of AQI in Beijing in 2014, which ranges from 0 to 500. Figure 9 B) shows the distribution of ΔAQI, where w=2. The majority of ΔAQIs fall in a range between -100 and 100, which is much narrower than [0, 500]. The upper bound of ΔAQI's range is [0, 500], no matter how big w is.

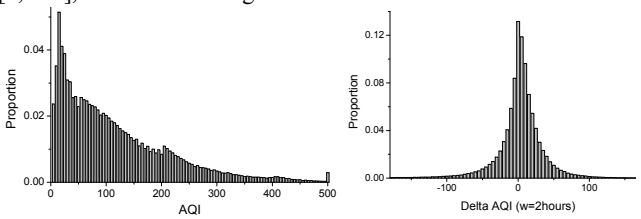


Figure 9. Distributions of AQI and ΔAQI in Beijing

The number of layers in the neural network depends on the scale of inputs, i.e. the number of non-empty regions, and the training data. For example, when there are 150 features and a one-year training dataset, we set a four-layer neural network (i.e. two hidden layers). By pairing the same inputs with the ΔAQIs of different time intervals we are going to predict, we train multiple spatial predictors corresponding to different future time intervals. This occurs in a similar fashion as with temporal predictors.

3.3 Prediction Aggregator

The prediction aggregator dynamically integrates the predictions that the spatial and temporal predictors have made for a location. The spatial and temporal predictors use non-overlapped features to predict the air quality of a location, offering different points of view (local and global) on the prediction. Sometimes, local information is more important than global information, e.g. when the air circulation between different places is weak. On the contrary, global dispersion may be a major factor in determining a place's air quality, e.g. when the wind speed is very high. As a result, we consider the current meteorology of the location, such as the wind speed, wind direction, humidity, and sunny/cloudy/overcast/foggy, to calculate a dynamic weight for the two predictions.

Specifically, we train a Regression Tree (*RT*) [5][9] to model the dynamic combination of these factors and predictions. A Regression Tree can be regarded as an integration of a Decision Tree and a Linear Regression. In general, it hierarchically partitions the data into groups based on some discriminative features and then learns a linear regression for each group of data in a leaf node. While the first step of a RT is similar to a Decision Tree, a RT can handle continuous and discrete features simultaneously. When handling continuous features, it uses the decrease in variance (somehow similar to information gain in a Decision Tree) in the data to determine partition thresholds. The feature that results in the most decreases in variance or information gain will be selected as the first node to partition the data into two parts. The process is performed in each part of the data iteratively, until some criteria have been satisfied, e.g. the depth of a tree or the number of instances in a leaf node.

To train such a regression tree, we deposit the predictions generated by the spatial and temporal predictors with the local meteorological data of the time interval in a feature set. The feature set is then paired with the corresponding ΔAQI (from the ground truth). The spatial and temporal predictors have been trained before we start training the prediction aggregator. Figure 10 presents a RT we train to predict the air quality of a station in Beijing, where an ellipse denotes a feature selected to partition the ΔAQIs; each square leaf node stands for a linear model (LM) that combines different features to calculate ΔAQI; the number associated with each edge is the threshold of a selected feature. All features have been normalized into [0,1]. For instance, when the value of a spatial predictor (*Spatial*) is smaller than 0.003 and the temporal prediction (*Temporal*) is greater than -0.08, we use LM4 to calculate ΔAQI. The weights of a feature in different LMs are different. For example, as presented in the right part of Figure 10, when wind speed is higher than 6.62, we select LM2, which gives temporal prediction a higher weight, to calculate ΔAQI. On the contrary, in LM3, spatial prediction is given a high weight. Each station has its own prediction aggregators that correspond to different time intervals to be predicted. So, the combination of spatial and temporal predictors changes over time and stations dynamically. The features that are not discriminative to determine the combination are ignored by the model automatically.

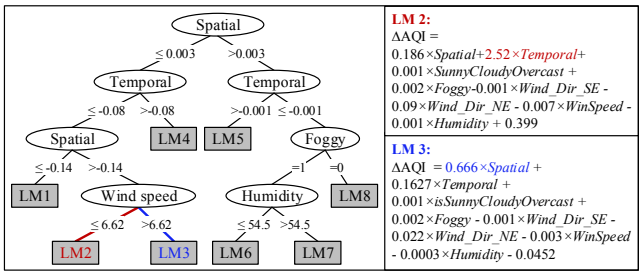


Figure 10. An example of Regression Tree

3.4 Inflection Predictor

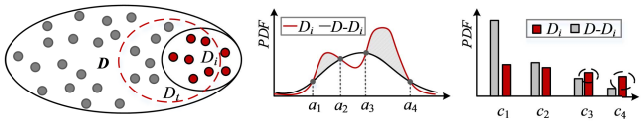
In some cases, the air quality of a location changes sharply in a few hours, which may be caused by a strong wind or a rain storm. Being able to predict such sudden changes is vital to informing people's decision making. However, the presence of such cases is very infrequent among entire observations; e.g. within one year of air quality data in Beijing, the presence of sudden drop instances is less than 1.1%. As a result, to predict them becomes almost impossible for the spatial and temporal predictors, which make a prediction based on the majority of observations.

To address this issue, we propose an inflection predictor, which is invoked to handle sudden changes when some criteria are satisfied. The predictor is built by the following four steps:

Step 1. Selecting the sudden drop instances D_i from historical data D: This step can be done by selecting the instances (from all the stations) whose AQI is bigger than 200 and decreases over a threshold in the next few hours, e.g. 50 in the coming one hour, or 100 in the coming two hours, or 150 in the coming three hours. In this study, we only focus on the sudden drop instances, as sudden increases of AQI are very rare in the real world. Even if a factory emission occurs, the air quality of the surrounding places usually becomes worse smoothly (because the volume of sources is much smaller than the capacity of the environment). This is also true when a foggy day is coming. Thus, such increasing cases can be handled by the spatial and temporal predictors. As depicted in Figure 11 A), the selected sudden drop instances are denoted by red points, while the rest are represented by gray points.

Step 2. Finding surpassing ranges and categories: We respectively calculate the distribution of each feature in the sudden drop instance set D_i and the entire dataset D. By comparing the two distributions of a feature, we find the ranges (for continuous features) or the categories (for discrete features) whose proportion in D_i is higher than the rest of the data (i.e. $D - D_i$). We call them surpassing ranges and categories. For example, as shown in Figure 11 B), the two curves denote the distributions of a continuous feature in D_i and $D - D_i$ respectively. We find that a feature's probability in ranges (a_1, a_2) and (a_3, a_4) in D_i is higher than $D - D_i$. Likewise, as illustrated in Figure 11 C), another discrete feature's proportions of

category c_3 and c_4 in D_i are higher than that of $D - D_i$. These may suggest that the features in these ranges or categories could be potential factors affecting sudden drop instances. Thus, some of them may be used as thresholds to invoke the inflection predictor. That is, once an instance has a feature's value falling in the surpassing ranges or categories, we send the instance simultaneously to the inflection predictor (besides the spatial and temporal predictors).



A) Select sudden drop instances D_i B) Distributions of a continuous feature C) Distributions of a discrete feature

Figure 11. Illustration of building an infection predictor

Step 3: *Selecting surpassing ranges and categories as thresholds*. While there are multiple surpassing ranges and categories, some of them may not really be discriminative enough (to be a threshold) to invoke the inflection predictor. To improve training quality, we need to find a set of surpassing ranges and categories as thresholds, with which we can retrieve as many instances from D_i as possible while involving the instances from $D - D_i$ as few as possible. Formally, the problem can be defined as finding a set of surpassing ranges (or categories) that maximizes Equation 1:

$$E = Max[(\frac{|x_1|}{|D_i|} - \frac{|x_2|}{|D-D_i|}) \times \frac{\Delta|x_1|}{\Delta|x_2|}; \quad (1)$$

Where $D_t = x_1 \cup x_2$ is a collection of instances retrieved by a set of surpassing ranges and categories; $x_1 \subset D_i$ is a collection of instances in D_t that belong to D_i; $x_2 \subset (D - D_i)$ is a collection of instances in D_t that belong to $D - D_i$; $|x|$ stands for the number of instances in collection x; $\Delta|x_1| = |x_1| - |x'_1|$ denotes the increment of instances (belonging to D_i) after adding a new surpassing range or category; x'_1 is the predecessor of x_1; likewise, $\Delta|x_2| = |x_2| - |x'_2|$ stands for the increment of instances belonging to $D - D_i$ after adding a surpassing range or category. The problem can be solved by using Simulated Annealing when there are many surpassing ranges and categories. Otherwise, we can find the most optimal combination through a brute force search.

Step 4. Training an inflection predictor with D_t: Using the thresholds selected from Step 3, we can retrieve a collection of instances D_t from the entire dataset D. We then train an inflection predictor based on D_t. Note that the selected surpassing ranges and categories are only used as thresholds to control when to invoke the inflection predictor. The features used in the inflection predictor to determine the specific drop values are the same as those of the temporal predictor. In implementation, the inflection predictor is based on a RT, which achieves a slightly better performance than using a linear regression, as some sudden drop instances may not follow a linear relationship with the features used. The output of the inflection predictor is a delta of AQI to be appended to the final result. As D_t contains instances from $D - D_i$, the prediction could be a non-dropping value. Thus, even when a non-sudden drop instance is sent to the inflection point, we can still predict them correctly.

Example: Table 1 shows an example of selecting surpassing ranges and categories based on the data of Beijing (from May 1st, 2014 to April 30th, 2015). Using the method proposed in Step 1, we find 3,184 sudden drop instances (D_i) from 292,167 instances D. By comparing the distributions of each features in D_i and D, we find six surpassing ranges and categories listed in the first column. The second column presents the percentage of instances (retrieved by only using a surpassing range or category) in D_i. The third column shows the percentage in $D - D_i$. These surpassing ranges and categories are sorted in a descending order by the ratio between the two percentages. The fourth column denotes the third part of Equation 1, and the fifth column presents the final score E. After adding the surpassing ranges or categories one by one (starting from the *WindSpeed*), the value of E increases until the fourth surpassing category is added. This is also the global maximum of E in all combinations of items shown in the first column. In the last two columns, the values shown at the i-th row are calculated based on the *first* i surpassing ranges and categories in the first column. For example, E=0.149 is the score when selecting *Wind Speed (13.9-max), Humidity (1-40)*, and *Downpour*. In the deployed system, we select the three surpassing ranges and category for Beijing. As long as a coming instance has a wind speed feature greater than 13.9m/s, and/or humidity lower than 40, and/or downpour, the instance will be sent to the inflection predictor (besides being sent to the spatial and temporal predictors).

Table 1. Example of selecting surpassing ranges/categories

Ranges/categories	$\|x_1\|/\|D_i\|$	$\|x_2\|/\|D-D_i\|$	$\Delta\|x_1\|/\Delta\|x_2\|$	E
WinSpeed:13.9-	**0.130**	**0.031**	**0.065**	**0.006**
Humidity:1-40	**0.380**	**0.173**	**0.128**	**0.026**
Downpour	**0.382**	**0.174**	**0.714**	**0.149**
Wind Northwest	0.478	0.263	0.078	0.017
Sunny	0.643	0.405	0.084	0.020
Moderate rainy	0.680	0.437	0.087	0.020

After using our method, D_t has a much higher presence of sudden drop instances than D, leading to a quality model predicting sudden drops online. For example, in Beijing the presence has been increased from 1.1% in D to 14.6% in D_t. Note that we would never be able to find some thresholds that can completely exclude instances from $D - D_i$ while embracing all instances from D_i. Thus, we cannot train the inflection predictor using D_i whose distribution differs from coming instances in online predictions.

4. EVALUATIONS

4.1 Settings

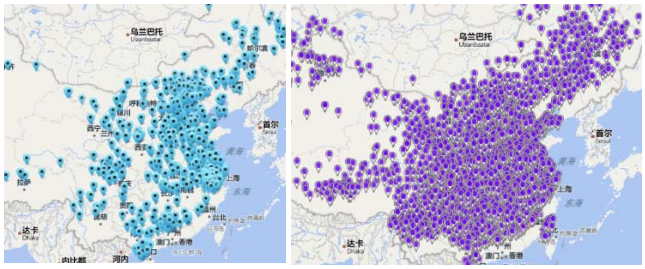
4.1.1 Datasets

Air quality data: Our system collects air quality data every hour from 2,296 stations in 302 Chinese cities. Figure 12 A) presents the geographical distribution of these stations, where each icon stands for a station. Each air quality instance consists of the concentration of six air pollutants: NO_2, SO_2, O_3, CO, PM2.5 and PM10. We convert these concentrations into corresponding (individual) AQIs for each air pollutant according to Chinese AQI standards (but without doing a 24-hour moving average over the AQIs). In total, over 12 million air quality instances have been collected from August 2012 to May 2015. As the cities are added into our system at different stages, the specific time spans of the AQI data in particular cities are different.

Meteorological data: The system collects meteorological data from 3,514 cities/districts/stations; Figure 12 B) shows these locations. Most major cities have a district-level (or even finer) granularity for the data, while small cities only have a city-level report. The location of a district (or city)-level meteorological report is represented by the geographical center of a district (or a city). Each meteorological record consists of sunny/cloudy/overcast/foggy/snowy/rainy, temperature, humidity, wind speed, and wind direction. Regarding rain and snow, there are different levels, such as minor rain, moderate rain, heavy rain and rainstorm. The meteorological data updates every hour, generating 16 million instances in total until April 30th, 2015.

Weather forecasts: The system collects weather forecasts for 2,612 cities/districts. The geographical granularity of a weather forecast is

very similar but slightly coarser than the meteorological data (a district-level at most) in some cities. We collect the forecast for the next three days for each update, which is usually segmented into multiple 3-hour (or six-hour) time intervals. A weather forecast for each time interval consists of sunny/cloudy/overcast/foggy/snowy /rainy, wind speed, and wind direction. The updating frequency of the forecasts varies by city (some cities are updated every 3 hours; some are 6 hours and 12 hours). In total, 203 million weather forecasts have been recorded until April 30th, 2015.



A) Air quality stations B) Meteorological sources

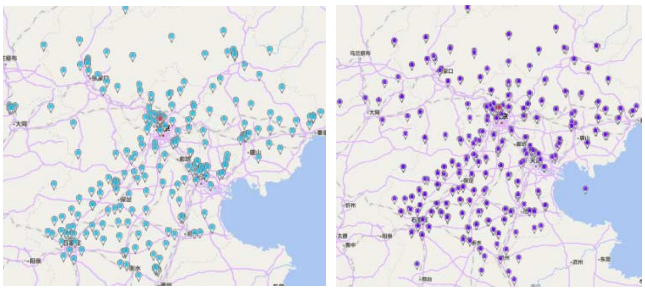
Figure 12 Sources of air quality and meteorological data

In this section, we present the evaluation of 4 major Chinese cities (Beijing, Tianjin, Guangzhou and Shenzhen) whose datasets have been detailed in Table 2. For example, Beijing has 36 air quality monitoring stations, 17 meteorological sources and 17 weather forecasting sources, respectively generating 278,085 air quality instances, 116,867 meteorological instances and 390,702 weather forecasts from May 1st, 2014 to April 30th, 2015. To predict the air quality of the 36 stations in Beijing, 233 air quality monitoring stations from 14 cities that are within 300km to Beijing are retrieved. Figure 13 A) shows the geographical distribution of these stations, which generate 1,272,979 air quality instances in the given time span. In addition, 177 meteorological sources from the 14 nearby cities are used in Beijing's evaluation, generating 1,006,814 meteorological instances. As the weather forecasting source is similar to the meteorological source, we only plot the geographical distribution of the latter in Figure 13 B).

Table 2. Some Details of Datasets

Datasets		Beijing	Tianjin	Guangzhou	Shenzhen
Time span		2014/5/1-2015/4/30	2014/5/1-2015/4/30	2014/5/1-2015/4/30	2014/5/1-2015/4/30
Nearby cities		14	17	19	19
AQI	In-city stations	36	27	42	11
	In-city instances	278,085	191,167	283,735	88,154
	Drop instances	3184	1945	134	8
	Ave. PM2.5	106.4	104.3	59.5	44.9
	Neighbor Sta.	233	267	145	148
	#. of instances	1,272,979	1,436,051	1,002,877	1,068,543
Meteorology	In-city sources	17	20	5	7
	In-city instances	116,867	106,614	30,305	55,632
	Nearby sources	177	195	115	122
	Near instances	1,006,814	1,108,873	626,418	665,463
Forecast	In-city sources	17	20	5	6
	In-city instances	390,702	361,624	106,380	51,870
	Nearby sources	184	182	110	114

In total, the data from another 39 cities has been involved in prediction for the four major cities. Since PM2.5 (Particulate Matter with a diameter smaller than 2.5 micrometers) is the most reported (and also the most difficult-to-predict) air pollutant, we focus the evaluation on PM2.5. Our method can be generally applied to other pollutants and countries. We partition the data into non-overlapped training and test data by a ratio of 2:1. For example, we select the data in March, June, September and December as the test set, and the other months as the train dataset. The data set has been released to the public in [17].



A) Nearby AQI stations for Beijing B) Meteorological sources for Beijing

Figure 13. Geo-distributions of the data sources for Beijing

Figure 14 shows the distributions of PM2.5'AQIs in the four cities, where the six AQI spans are defined by Chinese standards, respectively corresponding to *Good, Moderate, Unhealthy for Sensitive Group, Unhealthy, Very unhealthy and Hazardous*. As Beijing has the biggest population and the most complicated air quality, we focus on Beijing's data when comparing with different baselines, while showing overall results for the other three cities.

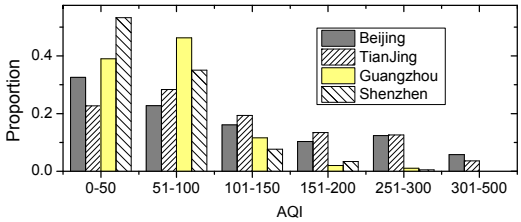


Figure 14. Distributions of AQI in PM2.5

4.1.2 Metrics and Ground Truth

We predict the air quality of a station as we can obtain the ground truth from its later readings. For the next 1-6 hours, we measure the prediction of each hour $\hat{y}_i$ against its ground truth y_i, calculating the accuracy according to Equation 2. With respect to the next 7-12, 13-24, 25-48 hours, we measure the mean of the predicted maximum and minimum values against the mean of the truth AQIs during the interval. As a result, we generate an accuracy for the four time intervals respectively at each station. We also calculate the absolute error of each time interval according to Equation 3, where n is the number of instances measured for a time interval.

$$p = 1 - \frac{\sum_i |\hat{y}_i - y_i|}{\sum_i y_i}; \quad (2)$$

$$e = \frac{\sum_i |\hat{y}_i - y_i|}{n}. \quad (3)$$

We aggregate the accuracy of the same time interval from all the stations in a city into a final result for the city. Finally, a city will have four overall accuracies and four absolute errors in 1-6, 7-12, 13-24, and 25-48 hours.

4.1.3 Baselines

We compare our method, entitled *FFA* (*TP+SP+PA+IP*), with four sets of baselines:

1) *ARMA*: Auto-Regression-Moving-Average (*ARMA*) is a well-known model for predicting time series data. ARMA predicts the air quality of a station solely based on the AQIs of the station. This baseline justifies the advantages of using weather forecasts and meteorological data.

2) This set of baselines feeds all features into a single model, e.g. linear regression (*LR_ALL*), neural network (*ANN_ALL*), and regre-

ssion tree (*RT_ALL*), without treating different features differently. Defining these baselines is to justify the advantages of using a combination of multiple models. As *LR*, *RT* and *ANN* have also been used in environmental science to predict air quality [4][5][6], surpassing this set of baselines also justifies our contribution over traditional approaches.

3) This baseline applies the classical weather forecasting model (*WFM*) to predict air quality. The results of *WFM* is generated by the Beijing Municipal Environmental Monitoring Center, published at http://zx.bjmemc.com.cn/ at 8am and 8pm every day.

4) The fourth set of baselines justifies the necessity of each component of our method. For example, if we do not use the inflection predictor (*IP*), or the prediction aggregator (*PA*).

4.2 Results

4.2.1 Results of Temporal Predictors

We first check if the features we feed into the temporal predictor are really useful. As shown in Table 3, by adding AQIs from the past three hours (*A*), time of day and day of the week (*T*), meteorological features (*M*), and weather forecasts (*F*) step by step, we see a clear improvement on the accuracy p and a decrease on the absolute error e at every future time interval we are going to predict. The results are generated by solely applying the temporal predictor to the test instances.

Table 3. Results of the temporal predictor in Beijing

Time	1-6h		7-12h		13-24h		25-48h	
Features	p	e	p	e	p	e	p	e
A	0.702	28	0.515	63.9	0.449	70.0	0.448	68.5
A+T	0.706	27.7	0.519	63.3	0.443	70.8	0.433	70.3
A+T+M	0.711	27.2	0.548	59.4	0.470	67.4	0.442	69.2
A+T+M+F	**0.713**	**27.0**	**0.560**	**57.9**	**0.477**	**66.5**	**0.461**	**66.8**

4.2.2 Results of Spatial Predictors

Table 4 presents the results of the spatial predictors respectively using (*ANN_Par*) and without using (*ANN_Raw*) the spatial partition and aggregation. According to the results of 7-12, 13-24, 24-48, the spatial partition and aggregation significantly improves the performance of the spatial predictor. Without this process, there are too many inputs for an *ANN*, leading to too many parameters in the model. Consequently, we cannot learn a set of accurate parameters for the *ANN* based on the limited training data. Additionally, the computational load of *ANN_Raw* is very heavy due to a large number of parameters involved. We also tested a linear regression model in the spatial predictor. In general, *LR* has a similar performance in predicting normal instances but less effective (2% lower) than *ANN* in dealing with sudden drops.

Table 4. The results of the spatial predictor in Beijing

Time	1-6h		7-12h		13-24h		25-48h	
	p	e	p	e	p	e	p	e
ANN_Raw	0.693	36.9	0.482	88.1	0.409	98.3	0.318	109.8
ANN_Par	**0.742**	**24.3**	**0.587**	**54.4**	**0.471**	**67.3**	**0.384**	**76.4**

4.2.3 Results of Prediction Aggregator

The results presented in Table 5 justify the advantages of the prediction aggregator (*PA*) which combines the predictions generated by the spatial and temporal predictors (*TP+SP+PA*). This table aggregates the accuracies and absolute errors of four different time intervals. First, *PA* improves the performance of individual spatial and temporal predictors, particularly in predicting sudden drop instances. Second, the combination of (*TP, SP, PA*) outperforms the second set of baselines: *LR_ALL* and *ANN_ALL,* which feed all features into a single model (*LR* or *ANN*).

Table 5. The results of prediction aggregator in Beijing

Methods	All Instances		Sudden Drops	
	p	e	p	e
Temporal Predictor (TP)	0.642	39.2	-0.314	125.2
Spatial Predictor (SP)	0.655	38.2	0.116	85.8
LR_ALL	0.667	36.7	0.015	94.4
ANN_ALL	0.647	39.0	0.150	82.0
TP+SP+PA	**0.670**	**36.4**	**0.173**	**80.5**

4.2.4 Results of Inflection Predictors

We learn the thresholds for the inflection predictor (*IP*) from the D_i of the entire dataset. The surpassing ranges and categories on the first three rows of Table 1 are selected as thresholds for Beijing. We then use these thresholds to find D_t from the training set and the test set respectively. As a result, 4,768 instances are used for training and 2,933 for testing, generating the results shown in Table 6. We find *RT* outperforms *LR* in predicting the sudden drops when used individually and in conjunction with *TP+SP+PA*. The *IP* also brings significant improvement over *TP+SP+PA*. The results of sudden drops in Table 5 are based on D_i while Table 6 is derived from D_t. Some drop instances are not retrieved by the thresholds.

Table 6. Results of the Inflection Predictor in Beijing

Metrics	Individually		***TP+SP+PA+IP***		***TP+SP+PA***
	LR	*RT*	*LR*	*RT*	
p	0.001	0.025	0.253	**0.262**	0.125
e	87.7	86.15	72.1	**72.9**	77.8

4.2.5 Overall results

Table 7 presents the overall results of different methods, where our method *FFA* outperforms all the baselines. *ARMA*-2 means an *ARMA* considering the recent 2 hours for a moving average. First, the meteorological data and weather forecasts bring improvements to air quality prediction. Second, as compared to *LR_ALL* and *ANN_ALL*, our method has a stronger capability of predicting the air quality of farther future and the sudden drops. The results justify the contribution of using a combination of three components rather than feeding all the features into a single model. The results also show that *ANN* is more capable of dealing with sudden changes than *LR*. In many real-world problems, we may not be able to get sufficient data to train a big model. Thus, a deep understanding of the data and the merit of different kinds of models is important.

Table 7. Comparison among different methods: in Beijing

Time	1-6h		7-12h		13-24h		25-48h		Sudden Changes	
Methods	p	e	p	e	p	e	p	e	p	e
AMRA-2	0.663	40.4	0.499	84.2	0.371	104.4	0.2	128.8	-0.622	179.1
AMRA-6	0.607	46.9	0.475	88.0	0.365	105.3	0.203	128.0	-0.523	170.6
LR_ALL	0.744	24.1	0.594	53.4	0.496	64.1	0.449	68.3	0.015	94.4
ANN_ALL	0.733	25.2	0.586	54.4	0.457	69.0	0.383	76.4	0.150	82.0
TP+SP+PA	0.75	23.6	0.601	52.4	0.498	63.9	0.444	69	0.173	80.5
FFA (TP+SP+PA+IP)	**0.749**	**23.7**	**0.601**	**52.4**	**0.498**	**63.9**	**0.444**	**69**	**0.262**	**72.1**

Third, the inflection predictor does not comprise the performance of our method, while significantly enhancing our method's capability of predicting sudden changes.

In Table 8, we compare our method *FFA* with *WFM*, which uses a weather forecasting model to predict air quality, during the time span: September 1st 2014 to April 30th 2015. The Beijing Municipal Environmental Monitoring Center (using *WFM*) only provides a district-level forecast for the next 12 hours, updating the forecast twice a day at 8am and 8pm. So, *FFA* has more accuracy predictions with a finer granularity and a farther forecasting period over *WFM*. In addition, *FFA* can update every hour, which indicates less online computational cost than *WFM*.

Table 8. Compare *FFA* with WFM in Beijing

Methods	1-6 hours		7-12 hours		Update	Grained
	p	*e*	*p*	*e*	Hours	Level
FFA	0.839	33.4	0.795	60.0	1	Station
WFM	0.761	49.6	0.777	65.3	12	District

Table 9 details the average absolute error at different time intervals and in different AQI ranges. For example, when predicting the air quality of the next 1 to 6 hours, the average absolute error for the air quality whose AQI falls in the range of [0, 50] is 17.5. Regarding the instances whose AQI falls into 50-100, the average absolute error is 21.2. According to the values shown in the last row, we can achieve an absolute error less than 38 when the real AQI is under 200. As the training instances falling into 300-500 are very small (refer to Figure 15), the error is relatively high in the range of [300-500].

Table 9. Average absolute error in different AQI ranges

Time	0-50	50-100	100-150	150-200	200-300	300-500
1-6	17.5	21.2	23.9	29.5	38.9	61.6
7-12	43.4	45.2	44.2	50.0	62.2	105.2
13-24	68.5	47.8	45.9	56.5	78.1	141.1
25-48	100.3	56.3	40.9	51.8	86.3	181.8
Total	35.2	30.7	30.5	37.3	51.1	88.6

Table 10 presents the results of our method in Beijing, Tianjin, Guangzhou and Shenzhen. Our method has a better performance in the latter two cities, as their air quality falls drastically in the range of [0,150] (see Figure 14) and the number of sudden changes is much smaller than Beijing and Tianjin (refer to Table 2). In short, their air quality is easy to predict. In such kinds of cities, our method does not show clear advantages beyond *TP*+*SP*+*PA*. We also compare our method with different baselines based on Tianjin's data, finding a similar trend there.

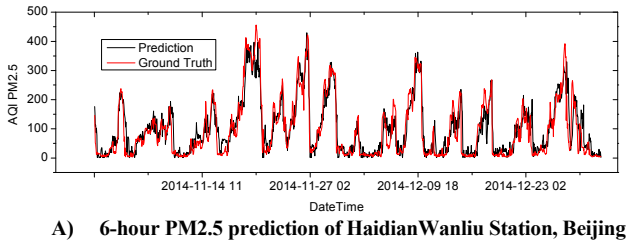
Figure 15 A) shows the prediction of our method at the next 6th hour against the ground truth in Beijing from Sep. 1, 2014 to Sep. 30, 2014. Figures 15 B), C) and D) present those of Tianjin, Guangzhou and Shenzhen. In general, Beijing and Tianjin have much more complicated air quality (changing over time) than Guangzhou and Shenzhen. Our model is very accurate in tracing the ground truth curves (including sudden changes) in the four cities. Figure 15 E) presents the average of the maxi-min predictions of our method for the next 7-12 hours against the ground truth (i.e. the gray area).

4.2.6 Efficiency and Resources

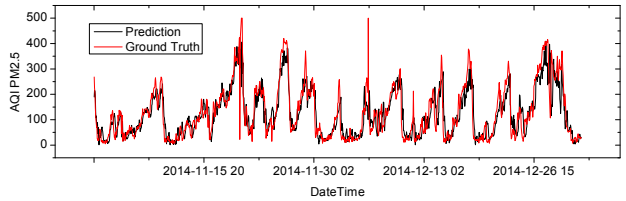
Table 11 presents the resources we use on the cloud (MS Azure) to enable the forecasting service. It also shows the time consumed by each component of our method to predict the air quality for a station at a time interval. On average, our method can generate a prediction in 3ms, finishing the forecast of the next 48 hours for a station in 36ms (recall that we have 12 models: 6 for the first six hours, 2 for each of the three time intervals). Given such a configuration, our service can answer over 40,000 request per hour. By adding more instances for the Azure Website, we easily upgrade our service to answer more queries.

Table 11. Configuration of Cloud and inference performance

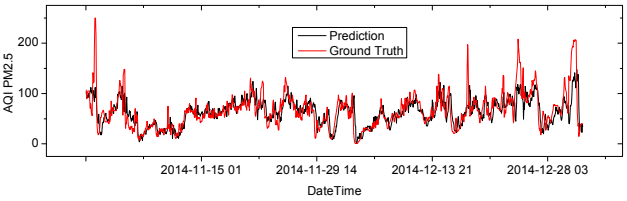
Services	Configurations	Models	Time (ms)
Azure WebSite	S2 Standard (2 cores, 3.5G Memo)	*Feature*	1.778
	3 instances	*TP*	0.010
Cloud Service	A1 (1 core, 1.75 GB Memory)	*SP*	0.108
	1 instance	*PA*	0.247
Database	Standard S0 (10 DTUs)	*IP*	0.508



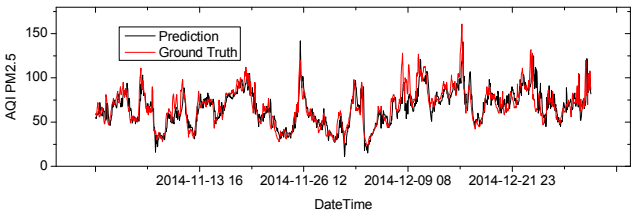
A) 6-hour PM2.5 prediction of HaidianWanliu Station, Beijing



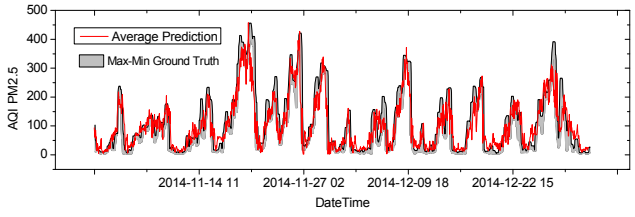
B) 6-hour PM2.5prediction of Dazhigu Station in Tianjin



C) 6-hour PM2.5 prediction of Luhu School, Guangzhou



D) 6-hour PM2.5 prediction of Nanyou station in Shenzhen



E) 7-12 hours PM2.5 prediction at HaidianWanliu in Beijing

Figure 15. *FFA*'s Predictions against ground truths

5. RELATED WORK

Existing air quality prediction methods in Environmental Science are usually based on classical dispersion models, such as Gaussian Plume models, Operational Street Canyon models, and Computational Fluid Dynamics [11].

Table 10. Overall results of our method in four major cities of China

Time	1-6h		7-12h		13-24h		25-48h		Sudden Changes	
Cities	*p*	*e*	*p*	*e*	*p*	*e*	*p*	*e*	*p*	*e*
Beijing	0.749	23.7	0.601	52.4	0.498	63.9	0.444	69	0.262	72.1
Tianjin	0.754	24.2	0.63	50.1	0.582	54.7	0.578	54.2	0.395	66.5
Guangzhou	0.797	11.3	0.717	22.4	0.676	25.2	0.644	27.5	0.572	45.9
Shenzhen	0.832	7.5	0.753	15.7	0.72	17.7	0.7	18.9	0.791	18.3

These models are in most cases a function of meteorology, street geometry, receptor locations, traffic volumes, and emission factors (e.g. g/km per single vehicle), based on a number of empirical assumptions and parameters that might not be applicable to all urban environments [11]. As these parameters are difficult to obtain precisely, the results generated by such kinds of models may not be very accurate [14]. We instead use a data-driven method to predict air quality rather than empirical model-based approaches.

Over the past decade, some statistic models, like linear regression, regression tree [5][9] and neural networks, have been employed in atmospheric science to do a real-time prediction of air quality [4][6] [7][12][13]. However, these methods simply feed a variety of features about a location into a single model to predict the future air quality of the location. Our method is distinguished from these approaches in three ways. First, besides the data of the location we predict, we also incorporate the data from other spatial neighbors (e.g. nearby stations), which send signals to the predictive model thereby significantly improving prediction accuracy. Second, we feed different data sources into different models, capturing the spatial correlation of air quality in different locations and the temporal dependency of air quality in a location simultaneously. These models are then aggregated organically to provide a more accurate prediction than solely feeding all the data into a single model. Third, our method is more capable of forecasting sudden changes in air quality than these other simple approaches. Being able to predict sudden changes is vital to informing people's decision making, but very difficult given such little presence in the entire body of observations.

More recently, there has been a trend of applying big data to solve urban challenges in the form of urban computing [15]. For example, in 2013, we used big data to infer the real-time and fine-grained air quality throughout an entire city [14][16]. Hsieh et al. [8] suggested the locations for air quality monitoring stations based on big data. Shang et al. [10] used GPS trajectories of sample of vehicles to infer the city-wide vehicular emissions. However, none of these technical works is concerned with forecasting air quality.

6. CONCLUSION

In this paper, we report on a real-time air quality forecasting system that uses data-driven models to predict fine-grained air quality over the following 48 hours. The system is based on a framework that connects the cloud with clients, collecting air quality, meteorological data and weather forecasts from over 3,000 sources (e.g. stations /districts/cities) in China. The mobile client, entitled Urban Air, and the website are public available at [1] and [2]. The forecasting function has also been deployed on Bing Map China at [3]. The system has also been deployed with the Chinese Ministry of Environmental Protection. We evaluate our predictive method with data from 43 cities, presenting the results of four major cities: Beijing, Tianjin, Guangzhou and Shenzhen. By combining four major components, consisting of temporal predictor, spatial predictor, prediction aggregator, and inflection predictor, our method outperforms four sets of baselines significantly, including the baseline approach using weather forecasting models to predict air quality. In general, our method can achieve an accuracy of 0.75 for the first 6 hours and 0.6 for the next 7-12 hours in Beijing. It predicts the sudden changes of air quality much better than baseline methods. With a very light resource strain on the cloud, on average, our method can generate predictions for the following 48 hours for a station in 36ms.

7. REFERENCES

[1] Urban Air Website: http://urbanair.msra.cn/

[2] Urban Air Windows Phone Client: http://www.windowsphone.com/en-us/store/app/urban-air/f36d5a33-2ccc-45f5-afd2-0c1afc5fc6dc

[3] Air Quality Forecasting on Bing Map: http://cn.bing.com/ditu/

[4] Air Quality Research Subcommittee of the Committee on Environment and Natural Resources CENR. Air Quality Forecasting: A Review of Federal Programs and Research Needs, June 2001

[5] Burrows, W.R., Benjamin, M., Beauchamp, S., Lord, E.R., McCollor, D., Thomson, B., 1995. CART Decision-Tree Statistical Analysis and Prediction of Summer Season Maximum Surface Ozone for the Vancouver, Montreal, and Atlantic Regions of Canada. J. Appl. Meteor. Climatol. 34.

[6] Donnelly, A., Misstear, B., Broderick, B. Real Time Air Quality Forecasting using Integrated Parametric and Nonparametric Regression techniques. *Atmospheric Environment* 103 (2015), pp. 53–65.

[7] Environmental Protection. Guideline for Developing an Ozone Forecasting Program. EPA-454/R-99-009. July 1999

[8] Hsieh, H. P., Lin, S. D., Zheng, Y. Inferring Air Quality for Station Location Recommendation Based on Big Data. In Proc. of *KDD 2015*, 2015.

[9] Lewis, R. J. An Introduction to Classification and Regression Tree (CART) Analysis. Presented at the 2000 Annual Meeting of the Society for Academic Emergency Medicine.

[10] Shang, J., Zheng, Y., Tong, W., Chang, E. and Yu, Y. "Inferring Gas Consumption and Pollution Emission of Vehicles throughout a City," In Proc. of *KDD'14,* pp. 1027-1036, 2014.

[11] Vardoulakis, S., Fisher, B. E. A., Pericleous, K., Gonzalez-Flesca, N. Modelling Air Quality in Street Canyons: A Review. *Atmospheric Environment* 37 (2003), pp. 155-182.

[12] Zhang, Y., Bocquet, M., Mallet, V., Seigneur, C., Baklanov, A., Real-time Air Quality Forecasting, Part I: History, techniques, and current status, Atmospheric Environment 60 (2012), pp. 632–655.

[13] Zhang, Y., Bocquet, M., Mallet, V., Seigneur, C., Baklanov, A., Real-time Air Quality Forecasting, Part II: State of the science, current research needs, and future prospects, *Atmospheric Environment* 60 (2012), pp. 656–676.

[14] Zheng, Y., Liu, F., Hsieh, H. P. U-Air: When Urban Air Quality Inference Meets Big Data. In Proc. of *KDD 2013*, pp. 1436-1444, 2013.

[15] Zheng, Y., Capra, L., Wolfson, O., Yang, H. "Urban Computing: Concepts, Methodologies, and Applications," *ACM Trans. Intelligent Systems and Technology*, vol. 5, no. 3, pp. 38-55, 2014.

[16] Zheng, Y., Chen, X., Jin, Q., Chen, Y. Qu, X., Liu, L., Chang, E., Ma, W.Y., Rui, Y., Sun, W., A Cloud-Based Knowledge Discovery System for Monitoring Fine-Grained Air Quality. MSR-TR-2014-40

[17] Released Data: http://research.microsoft.com/apps/pubs/?id=246398