Comparative Machine Learning Approaches for Prediction of Air Quality in Beijing

Andy Han[1], Tianyi Zhang[1]
Emory University
December 9, 2023

This study addresses Beijing's air quality challenge using the 'Beijing Multi-Site Air-Quality Data[2].' The primary goal is to evaluate and compare various machine learning models for accurately predicting air pollution levels in Beijing. This research emphasizes analyzing concentrations of PM2.5 and PM10, recognized as key air quality indicators. Supported by a comprehensive dataset, including observations from 12 monitoring stations over four years (March 2013 - February 2017), the authors aim to understand complex urban air pollution patterns and assess the predictive performance of different machine learning models.

[Data, Scripts, and Figures Created are accessible through the GitHub Repository[3]]

[1] Andy Han and Tianyi Zhang, both students at Emory University, have jointly authored this paper as part of their coursework for CS334 - Machine Learning (Section 2), under the guidance of Dr. Joyce Ho. Both authors have contributed equally to the research and writing of this paper. [see ***Appendix A:Team Member Contributions*** for detailed team dynamics and individual work distributions.]
[2] Dataset information: Beijing Multi-Site Air-Quality Data. Retrieved from UCI Machine Learning Repository. DOI: 10.24432/C5RK5G
[3] See ***Appendix C: GitHub Repository*** for more information.

Abstract

This research tackles the critical issue of air pollution in Beijing, focusing on the period from 2013 to 2017—a time characterized by consistently poor air quality. Advanced machine learning techniques are used to forecast the levels of particulate matter (PM2.5 and PM10) by examining spatial and temporal data collected from 12 monitoring sites across Beijing. The approach involves a precise process of data preprocessing, feature selection, and hyperparameter optimization, using the comprehensive Beijing Multi-Site Air-Quality Dataset. Through the application of sophisticated algorithms, such as Random Forest, Gradient Boosting, and Neural Network, the models demonstrate exceptional predictive accuracy, with high R^2 scores above 0.88 for both PM2.5 and PM10 predictions. These results not only surpass previous studies but also mark a significant advancement in air quality prediction. The findings are instrumental in addressing immediate public health concerns and in devising strategies for long-term environmental protection.

Keywords: *Air Pollution, Particulate Matter, Machine Learning, Predictive Modeling, Environmental Health, Public Health, Air Quality Forecasting, Beijing Air Quality.*

Introduction

The relentless march of technological progress over recent centuries has revolutionized the way of life, introducing groundbreaking innovations such as automobiles, trains, and airplanes. These developments have drastically reduced the time and cost associated with travel and transportation, yet they have not come without a cost. One of the most pressing challenges resulting from these advancements is the widespread issue of air pollution.

The air humans breathe, seemingly clear, is in fact a complex mixture of numerous microscopic particles that are invisible to the naked eye. The human respiratory system, despite its efficiency in filtering many of these particles, is not infallible, particularly in dealing with the more harmful varieties. As the size of these particles decreases, the respiratory system's ability to filter them diminishes. Notably, fine particulate matters such as P.M. 10 and P.M. 2.5 can bypass these natural defenses, infiltrating the lungs and causing significant health problems.

In recent years, the escalation of P.M. 2.5 and P.M. 10 levels—key indicators of air pollution—has become a source of major concern. This increase is largely attributed to heightened industrial activities, extensive use of fossil fuels, and burgeoning construction projects, leading to profound implications for both public health and the environment.

The World Health Organization (WHO) now equates the health risks associated with air pollution to those of other major global health threats. Having witnessed the impact of air pollution first-hand in China, where it has severely limited outdoor activities and disrupted everyday life, the authors understand the critical need for effective solutions.

In response, this study adopts a comparative machine learning approach to develop a predictive model for P.M. 2.5 and P.M. 10 levels. The methodology for this study encompasses a wide range of machine learning models and evaluation metrics, designed to provide a potent tool in the fight against air pollution and to foster improved community health and environmental well-being.

Related Works

The study of air pollution, while not extensively covered in the research, has seen some noteworthy research that offers valuable insights into the topic and suggests methodologies that can be adapted for the research purposes.

- ***Spatiotemporal Variations of Air Pollutants and Ozone Prediction*** : This research explores the relationship between spatiotemporal variations and air pollution. It employs Random Forest and Decision Tree Regression models to predict air pollution levels. The methodology used in this study to analyze basic spatial and temporal relationships with air pollution forms a foundational basis for this research. Building upon their work, the authors aim to include a wider array of variables and a more extensive set of machine learning algorithms, which could potentially enhance the accuracy of the models used in the research (Lyu et al., 2022).
- ***A Machine Learning Approach to Evaluate Beijing Air Quality***: Yang's research focuses on identifying factors that might affect air quality in Beijing. One of the key methodologies adopted in this study is feature engineering, which is used to facilitate faster hyperparameter tuning and improve prediction accuracy. This approach has been incorporated into this research as well, particularly in the preprocessing phase of the dataset. Yang's work provides a significant reference point for this study, especially in understanding how feature engineering can be effectively utilized in air quality research (Chen, 2019).

Both these studies contribute to the field by applying machine learning techniques to understand and predict air quality. Their methodologies, particularly in terms of machine learning model selection and feature engineering, have informed and influenced the approach taken in this current research. By extending these methodologies and incorporating a broader set of variables and machine learning algorithms, this research aims to further the understanding of air quality prediction and its influencing factors.

Dataset Information

The study utilizes the "Beijing Multi-Site Air-Quality Data," contributed by Song Chen, sourced from the UCI Machine Learning Repository. This dataset encompasses data from 12 monitoring sites across Beijing, collected over the period from March 1st, 2013, to February 28th, 2017. It comprises a total of 420,768 instances, with each of the 12 datasets containing 35,064 instances. The dataset includes instances of missing data (marked with NA values) and categorical variables.

The dataset employed in this study is comprehensive, encompassing a variety of environmental parameters. It is structured as follows:

1) **Temporal Markers:** These include 'Year', 'Month', 'Day', and 'Hour' of data collection, all integers, representing the precise time point of each measurement.

2) **Air Quality Indicators:**

- *Particulate Matter Concentrations:* PM2.5 and PM10 levels are recorded in micrograms per cubic meter (µg/m^3).
- *Gaseous Pollutants:* This includes sulfur dioxide (SO_2), nitrogen dioxide (NO_2), carbon monoxide (CO), and ozone (O_3) concentrations, also measured in micrograms per cubic meter (µg/m^3).

3) **Meteorological Data:**

- *Temperature (TEMP):* Recorded in degrees Celsius (°C).
- *Atmospheric Pressure (PRES):* Measured in hectopascals (hPa).
- *Dew Point Temperature (DEWP):* Noted in degrees Celsius (°C).
- *Precipitation (RAIN):* Documented in millimeters (mm).

4) **Wind Characteristics:**

- Direction (Wd): Categorized based on the compass direction.
- Speed (WSPM): Wind speed, measured in meters per second (m/s).

5) **Location Data:** 'Station', a categorical variable, identifies the air-quality monitoring site from which the data was collected.

6) **Index:** Each record in the dataset is uniquely identified by a row number ('No'), serving as an integer index.

Exploratory Data Analysis

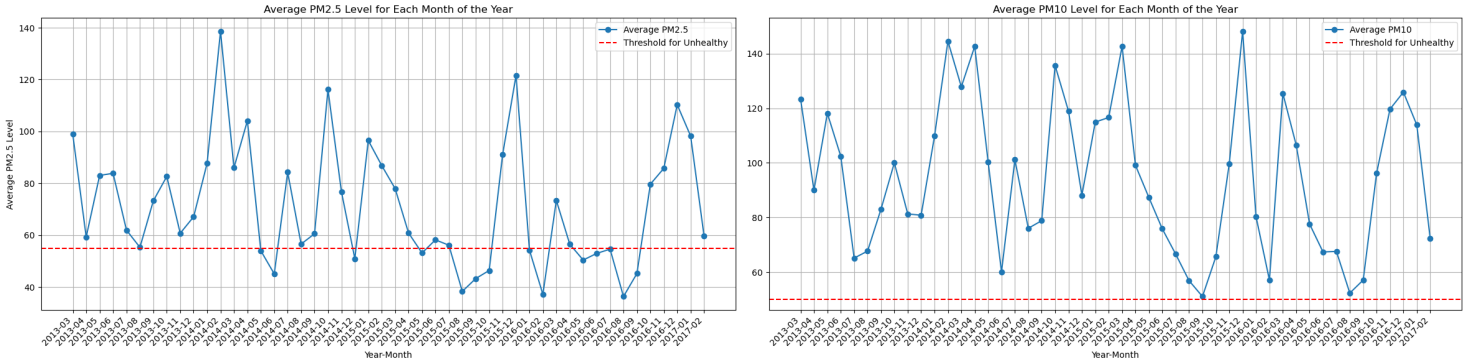

Figure 1, 2: Monthly average P.M.2.5 and P.M.10 level for 2013-2017. [4]

The monthly average PM2.5 and PM10 values from 2013-2017 are graphically depicted, revealing a stark portrayal of Beijing's air quality challenges. The inclusion of a red dotted line, representing the EPA's threshold for unhealthy air pollution, further emphasizes the severity of the situation (AirNow.gov). The finding is alarming: almost every month during this period, pollution levels surpassed the threshold, highlighting a chronic air quality crisis.

Moreover, this analysis aligns with research indicating that seasonal factors, particularly during winter, exacerbate pollution in Beijing. The increased use of fireworks and heightened travel during cultural events like the Spring Festival further contribute to these spikes, as shown in studies “Analysis of Pollution Characteristics,Meteorological Impact,and Forecast Retrospective During the Spring Festival and the Lantern Festival in “2+26” Cities” (Zhu et al., 2022). This research underscores the intricate relationship between cultural practices, seasonal changes, and air pollution, necessitating targeted

[4] For visual insights and detailed figures related to the "Prediction of Air Quality in Beijing" study, please visit the GitHub repository. You can find the relevant illustrations and graphs in the *Figures_for_paper* folder.

environmental policies and public awareness campaigns to allow citizens to take preventive actions against the heavy air pollution in a timely manner.

Methodology

A. Data Preprocessing.

The dataset, comprising 12 distinct datasets from various sites, is integrated into a single dataset. The preprocessing steps included:

1. **One-Hot Encoding:** Categorical features, particularly wind direction and station names, are converted using one-hot encoding. Different station names are manually assigned unique values.
2. **Imputation:** Replacing null values with the mean value of the column it belongs to. By conducting imputation, the machine learning algorithms are able to run on the dataset while preserving the same mean for each feature.
3. **Feature Engineering:**
    a. **Join Features:** The 'year', 'month', and 'date' columns are combined into a single 'year-month-day' format (e.g., 2013-03-01).
    b. **Add Features:** New features are created, including 'holiday_proximity', 'is_holiday', 'day_of_week', 'Station', 'PM2.5_lag_1', and 'PM10_lag_1'.
        - The 'is_holiday' feature assesses if a date falls on a public holiday, and 'holiday_proximity' gauges the closeness to the nearest holiday. This consideration is essential, as research indicates a correlation between holidays and increased air pollution levels (Hua et al., 2021).
        - The inclusion of 'lag' features, representing the previous day's PM2.5 and PM10 values, accounts for their potential temporal influence on air quality metrics. The approach in this study mirrors the methodology of AirNow, a partner of the U.S. Environmental Protection Agency, which forecasts the average 24-hour concentration for the following day, thus validating the use of prior day data for predicting subsequent air pollution levels (AirNow.gov).
        - The 'Station' feature converts the data from all 12 monitoring sites into a numerical format, assigning each a unique integer value. This allows for more nuanced spatial analysis and helps identify pollution trends specific to different areas in Beijing.
        - Incorporating the 'day_of_week' feature into this study acknowledges the influence of weekly cycles on air pollution levels, as analyzed in the He's article (2023). This study reveals the inclusion of the 'day_of_week' feature to account for weekly fluctuations in air pollution, essential for accurate forecasting in urban areas.
    c. **Drop Features:** Unnecessary features like 'No' and the original 'year-month-day' are removed.
4. **Standardization:** Numerical features are standardized using the StandardScaler() from the sklearn library to have a mean of 0 and a standard deviation of 1.
5. **Extract the target variables:** In this research, the primary focus is on predicting air quality, specifically the concentrations of particulate matter. To this end, we have designated two key target variables from the comprehensive final feature set: PM2.5 and PM10.

B. Final Feature Set.

- Target Variables:

1) PM2.5: PM2.5 concentration (μg/m^3).
2) PM10: PM10 concentration (μg/m^3).

- Temporal and Pollution Data:

3) Hour: Hour of data recording
4) SO_2: SO_2 concentration (μg/m^3).
5) NO_2:NO_2 concentration (μg/m^3).
6) CO: CO concentration (μg/m^3).
7) O_3: O_3 concentration (μg/m^3).

- Meteorological Data:

8) TEMP: Temperature (°C).
9) PRES: Pressure (hPa).
10) DEWP: Dew point temperature (°C).
11) RAIN: Precipitation (mm).
12) WSPM: Wind speed (m/s).

- Wind Characteristics:

13) wd_E: Wind Direction East
14) wd_ENE: Wind Direction East North East
15) wd_ESE: Wind Direction East South East
16) wd_N: Wind Direction North
17) wd_NE: Wind Direction North East
18) wd_NNE: Wind Direction North North East
19) wd_NNW: Wind Direction North North West
20) wd_NW: Wind Direction North West
21) wd_S: Wind Direction South
22) wd_SE: Wind Direction South East
23) wd_SSE: Wind Direction South South East
24) wd_SSW: Wind Direction South South West
25) wd_SW: Wind Direction South West
26) wd_W: Wind Direction West
27) wd_WNW: Wind Direction West North West
28) wd_WS: Wind Direction West South
29) Wd_NaN: No Wind

- Special Features:

30) Holiday_proximity: Number of days of difference to the closest holiday
31) Is_holiday: If that day is holiday
32) Day_of_week: The day of the week
33) Station: Station number

The feature selection for the study is carried out following comprehensive preprocessing steps. All features, except for the station identifiers, have been standardized and are represented as floating numbers. This standardization is essential for maintaining consistency across the dataset, allowing for more effective processing by machine learning algorithms.

The 'Station' feature, initially categorical, has been converted to a numerical format by assigning each of the 12 monitoring stations a unique number from 1 to 12. This transformation aids in integrating spatial aspects into the predictive model, thereby enhancing its ability to forecast air quality with greater precision.

C. Correlation between features.

Upon standardizing the features and appropriately encoding categorical variables, Pearson correlation heatmaps are used to investigate the interdependencies among the variables. The Pearson correlation coefficient provides a measure of the linear correlation between variables, with values ranging from -1 to +1. A value of +1 indicates a perfect positive correlation, 0 indicates no correlation, and -1 indicates a perfect negative correlation.

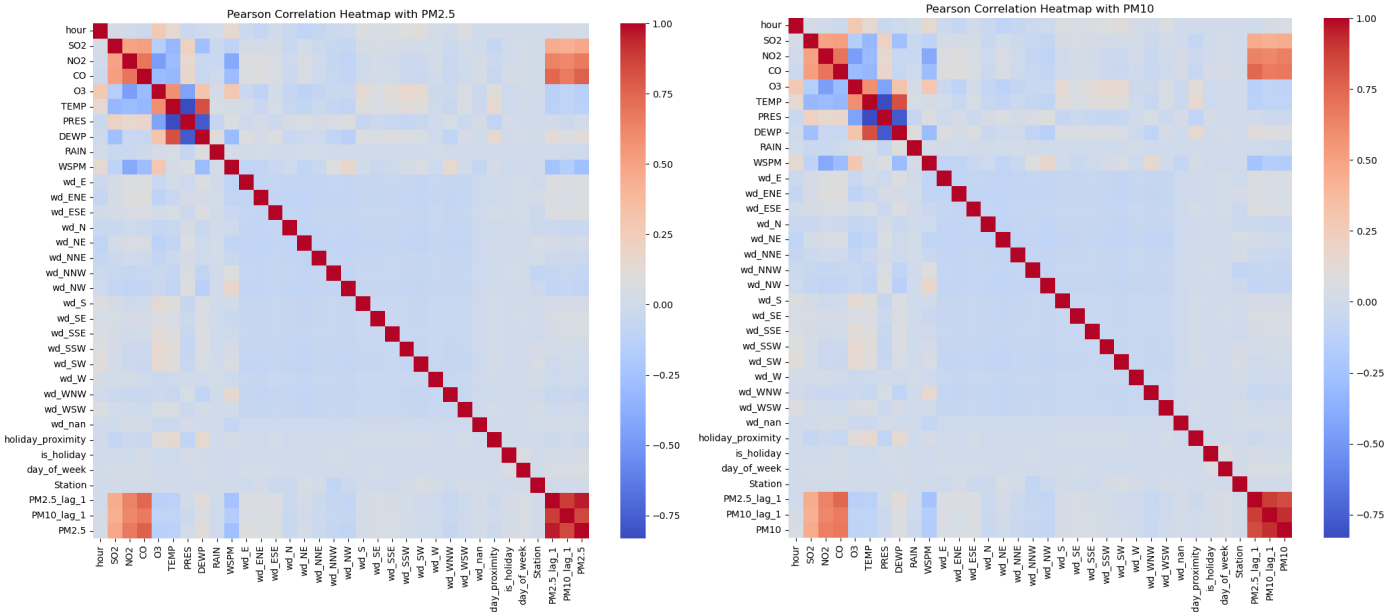

Figure 3, 4: Pearson Correlation Heatmaps for Target Variables and other Features[5]

The Pearson correlation heatmaps serve as an integral component of our exploratory data analysis, offering a preliminary glimpse into the linear associations between environmental variables within the air quality dataset. The heat maps show a spectrum of correlations that range from strongly positive to strongly negative, with the coefficients providing a numerical measure of the strength and direction of these relationships (Waskom, 2021).

The notable positive correlations among various pollutants suggest the possibility of shared emission sources or congruent atmospheric processes that affect their concentration levels in a similar manner. In contrast, the inverse relationship observed between certain meteorological conditions—such as

[5] For visual insights and detailed figures related to the "Prediction of Air Quality in Beijing" study, please visit the GitHub repository. You can find the relevant illustrations and graphs in the *Figures_for_paper* folder.

temperature and wind speed—and particulate matter concentrations indicates environmental influences that may facilitate the dispersion of these pollutants, thus impacting air quality.

However, the inherent limitation of the Pearson correlation coefficient in detecting only linear correlations necessitates caution in drawing conclusions from these patterns. While these coefficients are useful for flagging potential predictor variables for air quality, they do not confirm causality or capture more complex, non-linear interactions that may be at play.

Consequently, the methodological approach dictates that the full suite of features is retained from the analytical arsenal as the research advances to more sophisticated stages of modeling. The retention of all features is strategic, allowing the use of a variety of computational techniques, including machine learning algorithms that are adept at modeling the non-linear and interactive effects that simpler statistical measures might overlook. This comprehensive approach ensures that the predictive models are robust, nuanced, and capable of providing insightful forecasts that can inform environmental policy and public health initiatives.

D. Train-Test Split.

For the study, the dataset is partitioned utilizing sklearn's train_test_split function. 80% of the data for training purposes is allocated and the remaining 20% is reserved for testing. This split ratio is strategically chosen considering the extensive size of the dataset, which exceeds 400,000 entries. By assigning a larger portion to the training subset, the authors ensure that the models are trained on a robust and comprehensive set of data points, thereby enhancing the learning and generalization capabilities.

Importantly, the test set is specifically composed of the latter 20% of the data. This approach is adopted to maintain the chronological integrity of the dataset, ensuring that the temporal progression inherent in the data is accurately reflected in the models' evaluation phase. The aim is to assess the models' predictive performance in a realistic and sequentially consistent scenario, mirroring real-world conditions where future data points are predicted based on past and present information.

E. Model Selection.

In this comparative study, a diverse array of machine learning algorithms is employed to identify the most effective models for predicting the target variable, PM2.5 and PM10 concentrations. The selection covers a broad spectrum of approaches, ranging from fundamental linear models to more sophisticated ensemble methods and neural network architectures. Each selected model possesses distinct characteristics and underlying assumptions, rendering them uniquely suited for various data types and predictive scenarios.

The primary objective of this study is to conduct a comprehensive comparison of these models, focusing on their accuracy, generalizability, and computational efficiency. By exploring this wide range of models, this research provides an in-depth analysis of their performance in the context of air quality forecasting, thus contributing valuable insights into the field of environmental data analysis.

1. **Random Forest** builds multiple decision trees and merges them together to get a more accurate and stable prediction. The prediction of the random forest regressor for an input vector $\mathbf{x}$ is the average of the predictions from all individual trees in the forest, where t_i is the prediction of the i^{th} tree (Pedregosa et al., 1970).

$$\hat{y}(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} t_i(\mathbf{x})$$

2. **Gradient Boosting** builds an additive model in a forward stage-wise fashion. Each regression tree h is fitted on the negative gradient of the given loss function L .

$$F_m(x) = F_{m-1}(x) + \rho_m h_m(x)$$

where F_m is the model at iteration m, and ρ_m is the learning rate (Pedregosa et al., 1970).

3. **Decision Tree** is a set of if-then-else decision rules based on the features of an object. The value $\hat{y}$ at each leaf node is the mean of the values of the instances that reach this leaf.

$$\hat{y} = \frac{1}{n} \sum_{i \in D_{\text{leaf}}} y_i$$

where D_{leaf} is the set of data points that reach the leaf, and y_i are their values (Pedregosa et al., 1970).

4. **Ridge Regression** adds ℓ_2 regularization to linear regression, minimizing the objective function.

$$\min_{\mathbf{w}} \left\{ \sum_{i=1}^{n} (y_i - \mathbf{w}^\top \mathbf{x}_i)^2 + \alpha \|\mathbf{w}\|^2 \right\}$$

where $\mathbf{w}$ is the coefficient vector, $\mathbf{x}_i$ are the feature vectors, y_i are the target values, and α is the regularization strength (Pedregosa et al., 1970).

5. **Lasso** adds ℓ_1 regularization to linear regression .

$$\min_{\mathbf{w}} \left\{ \frac{1}{2n} \sum_{i=1}^{n} (y_i - \mathbf{w}^\top \mathbf{x}_i)^2 + \alpha \|\mathbf{w}\|_1 \right\}$$

Lasso tends to produce some coefficients that are exactly zero, thus performing feature selection (Pedregosa et al., 1970).

6. **Elastic Net** combines ℓ_1 and ℓ_2 regularization (Pedregosa et al., 1970).

$$\min_{\mathbf{w}} \left\{ \frac{1}{2n} \sum_{i=1}^{n} (y_i - \mathbf{w}^\top \mathbf{x}_i)^2 + \alpha\rho \|\mathbf{w}\|_1 + \frac{\alpha(1-\rho)}{2} \|\mathbf{w}\|^2 \right\}$$

7. **Bayesian Ridge** introduces a probabilistic model of the linear regression problem. The likelihood of observing $\mathbf{y}$ given $\mathbf{X}$ is assumed to be Gaussian .

$$p(\mathbf{y}|\mathbf{X}, \mathbf{w}, \alpha) = \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{w}, \alpha)$$

where $\mathcal{N}$ is the Gaussian distribution, and \alpha is the precision of the distribution (Pedregosa et al., 1970).

8. **Polynomial Regression** fits a nonlinear relationship between x and y.

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_n x^n + \varepsilon$$

where β_i are the coefficients, and ε is the error term (Pedregosa et al., 1970).

9. **SGD** (Scholastic Gradient Descent) minimizes an objective function by updating the parameters in the opposite direction of the gradient.

$$\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla Q(\mathbf{w})$$

where η is the learning rate, and $\nabla Q(\mathbf{w})$ is the gradient of the objective function (Pedregosa et al., 1970).

10. **Neural Network** with one hidden layer can be represented as.

$$\hat{y} = f(\mathbf{W}_2 f(\mathbf{W}_1 \mathbf{x} + \mathbf{b}_1) + \mathbf{b}_2)$$

where f is an activation function, $\mathbf{W}_1, \mathbf{W}_2$ are weight matrices, and $\mathbf{b}_1, \mathbf{b}_2$ are bias vectors (Pedregosa et al., 1970).

F. Model Evaluation Metrics and Efficiency Reporting.

In assessing the performance of the selected machine learning models, three key metrics are chosen: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2). Additionally, this research will consider the time efficiency of each model in terms of training and prediction.

- Mean Absolute Error (MAE): MAE measures the average magnitude of the errors in a set of predictions, without considering their direction. It is calculated as the average of the absolute differences between the predicted values and observed values. MAE is a linear score, meaning all individual differences are weighted equally in the average.

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

where y_i is the true value and $\hat{y}_i$ is the predicted value (Pedregosa et al., 1970).

- Root Mean Squared Error (RMSE): RMSE is a quadratic scoring rule that measures the average magnitude of the error. It is the square root of the average of squared differences between prediction and actual observation. RMSE gives a relatively high weight to large errors, making it useful when large errors are particularly undesirable (Pedregosa et al., 1970).

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

- R-squared (R^2): R^2, also known as the coefficient of determination, is a statistical measure that represents the proportion of the variance for the dependent variable that is explained by the independent variables in a regression model. It provides an indication of goodness of fit and therefore a measure of how well unseen samples are likely to be predicted by the model.

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

where $\bar{y}$ is the mean of the observed data (Pedregosa et al., 1970).

In addition to these performance metrics, each model's efficiency will be evaluated in two aspects:

- **Training Time:** The time taken to train the model on the training dataset.
- **Prediction Time:** The time required for the model to make predictions on new data.

Both training and prediction times are crucial in real-world applications, especially for large datasets or in scenarios requiring fast responses. By reporting these metrics, this research aims to provide a comprehensive overview of each model's strengths and weaknesses, balancing accuracy with computational efficiency.

G. Comparative Model Selection and Hyperparameter Optimization.

For the comparative analysis, this research adopts a structured approach to model selection and hyperparameter optimization:

Data Subset for Hyperparameter Tuning:

In the approach to model selection and hyperparameter tuning, 10% of the training data is randomly selected from the training dataset. This decision is made to streamline computational efficiency while maintaining a representative sample of the dataset's broader characteristics. Using a subset of the training data for this phase effectively mitigate the risk of data leakage.

Restricting the hyperparameter tuning to a portion of the training set ensures that the models are not inadvertently exposed to the test set. This safeguard maintains the integrity of the evaluation process, ensuring that the models are tested on completely unseen data, which is a critical aspect of assessing their real-world applicability and robustness.

Cross-Validation and Grid Search Methodology:

A combination of cross-validation techniques and grid search methodology are employed to ensure the robustness and reliability of each machine learning model. 5-fold cross-validation is chosen to evaluate model performance across various data subsets, thereby mitigating the potential for bias due to peculiarities in specific data sets.

In parallel, a comprehensive grid search is conducted to systematically explore and identify the optimal hyperparameters for each model. This process involves an extensive exploration of different parameter combinations to refine the models, thereby enhancing their accuracy and generalizability in predicting air quality. For a detailed list of the parameter combinations assessed during this study, please refer to Appendix B[6].

This combined approach of employing both 5-fold cross-validation and grid search ensures a careful examination of each model's capabilities, significantly improving the reliability and effectiveness of the predictive analysis.

[6] Due to the extensive nature and comprehensive scope of the hyperparameter combinations explored in this study, these details are not included in the main text to maintain brevity and focus. For a complete and detailed overview of the hyperparameters and the grid search methodology employed, readers are directed to ***Appendix B: Set of Hyperparameters Evaluated in Grid Search***.

H. Set of Optimal Hyperparameters

Model	Best Params For PM2.5	RMSE	MAE	R2
RandomForest	{'max_depth': None, 'min_samples_leaf': 4, 'min_samples_split': 2, 'n_estimators': 200}	0.154591133	0.072724676	0.975912909
GradientBoosting	{'learning_rate': 0.05, 'max_depth': 7, 'n_estimators': 400}	0.125253725	0.084271807	0.984187647
DecisionTree	{'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 4}	0.206854405	0.119938085	0.956873448
Ridge	{'alpha': 10}	0.253697519	0.141181542	0.935129442
Lasso	{'alpha': 0.1}	0.282042271	0.170473332	0.919824145
ElasticNet	{'alpha': 0.1, 'l1_ratio': 0.2}	0.272146933	0.162695176	0.925351329
BayesianRidge	{'alpha_1': 1e-06, 'alpha_2': 0.0001}	0.253697048	0.141132996	0.935129683
PolynomialRegression	{'linear__fit_intercept': True, 'poly__degree': 2}	0.234404225	0.132055708	0.944620878
SGD	{'max_iter': 2000, 'penalty': 'elasticnet', 'tol': 0.0001}	0.254494475	0.141699625	0.934721238
NeuralNetwork	{'hidden_layer_sizes': (50,), 'max_iter': 1000}	0.228392714	0.133385692	0.947424951

Figure 5: Optimal Hyperparameters and Reported Performance Metrics (PM 2.5)[7]

Model	Best Params For PM10	RMSE	MAE	R2
RandomForest	{'max_depth': 20, 'min_samples_leaf': 4, 'min_samples_split': 4, 'n_estimators': 150}	0.240513559	0.115065856	0.942646652
GradientBoosting	{'learning_rate': 0.1, 'max_depth': 5, 'n_estimators': 400}	0.226052676	0.150017158	0.949336061
DecisionTree	{'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 8}	0.313194844	0.183141248	0.90274569
Ridge	{'alpha': 10}	0.373921213	0.201154563	0.861375544
Lasso	{'alpha': 0.1}	0.396409518	0.231322632	0.844199878
ElasticNet	{'alpha': 0.1, 'l1_ratio': 0.2}	0.386138666	0.217024135	0.852168756
BayesianRidge	{'alpha_1': 1e-06, 'alpha_2': 0.0001}	0.373921101	0.201146296	0.861375627
PolynomialRegression	{'linear__fit_intercept': False, 'poly__degree': 2}	0.34980988	0.197280288	0.878676819
SGD	{'max_iter': 2000, 'penalty': 'l1', 'tol': 0.0001}	0.375369104	0.202371434	0.860299907
NeuralNetwork	{'hidden_layer_sizes': (50,), 'max_iter': 1000}	0.342326906	0.198844179	0.883811882

Figure 6: Optimal Hyperparameters and Reported Performance Metrics (PM 10)[8]

The figures present hyperparameters fine-tuned through GridSearchCV, which employs cross-validation to minimize the negative mean squared error. This approach ensures that the selected parameters offer the most accurate predictions and meaningful insights, according to the performance metrics reported.

I. Final Model Training

After identifying optimal hyperparameters, each model undergoes a comprehensive training process using the full dataset with the optimal set of hyperparameters respectively. This step not only evaluates their predictive accuracy using RMSE, MAE, and R-squared Score on both training and test sets but also measures their time efficiency. Specifically, the training time records the duration taken to fit the model to the training data, while the testing time measures how quickly the model can make predictions on new data. These time metrics are crucial for understanding the models' practical applicability in real-world scenarios, where both accuracy and speed are essential.

J. Accessing Supplementary Materials

For a deeper insight into the research process, scholars and interested parties are invited to explore the GitHub repository. The repository can be accessed at: GitHub - Prediction of Air Quality in Beijing.

This online resource provides a comprehensive array of data files, encompassing each stage of the analysis, as well as the complete suite of Python scripts employed. These materials serve as an extension to this paper, offering an intricate understanding of the methodologies applied.

Results

Upon completing the training with optimal hyperparameters, the models are evaluated based on their RMSE, MAE, and R-Squared scores to establish a hierarchy of performance. The optimal models are

[7] For visual insights and detailed figures related to the "Prediction of Air Quality in Beijing" study, please visit the GitHub repository. You can find the relevant illustrations and graphs in the *Figures_for_paper* folder.
[8] Same as *footnote 6*.

characterized by the lowest RMSE and MAE values, indicative of minimal prediction errors, alongside the highest possible R-Squared values, signifying a strong explanatory power over the variance in the data. These criteria are essential for determining the models that offer the most precise and consistent predictions.

Model for PM2.5	Train RMSE	Train MAE	Train R2	Test RMSE	Test MAE	Test R2	Train Time (s)	Test Time (s)
RandomForest	0.119306832	0.074918249	0.985639812	0.220476012	0.122483236	0.953038877	554.5250962	1.412655592
GradientBoosting	0.233415432	0.128602351	0.945034697	0.229648625	0.129230497	0.949050083	120.1994674	0.100121737
DecisionTree	0.223127758	0.127149879	0.949773063	0.249965129	0.134683906	0.939636469	2.35859704	0.008510828
Ridge	0.260937698	0.142244683	0.931308477	0.250559511	0.14170224	0.939349056	0.064146996	0.00200963
Lasso	0.288530145	0.171314565	0.916013036	0.278798626	0.170217157	0.924907418	0.116685629	0.002500772
ElasticNet	0.278877738	0.164016317	0.921538395	0.270510861	0.163508917	0.929305571	0.136241198	0.00199914
BayesianRidge	0.260937694	0.142239881	0.931308479	0.250558336	0.141696836	0.939349625	0.161545753	0.002500772
PolynomialRegression	0.244551776	0.133617653	0.939664747	0.23704808	0.135312444	0.945713897	5.429141998	0.283946037
SGD	0.261568497	0.142959719	0.930975961	0.250982858	0.142344977	0.939143931	0.494775057	0.002500057
NeuralNetwork	0.235431365	0.132613485	0.944081162	0.232152326	0.134387232	0.947933084	44.53309202	0.017954826

Figure 7: Testing Result for Each Model (PM2.5)[9]

Model for PM10	Train RMSE	Train MAE	Train R2	Test RMSE	Test MAE	Test R2	Train Time (s)	Test Time (s)
RandomForest	0.122704188	0.066607577	0.984835225	0.326465102	0.187501715	0.896370887	964.4520311	2.911074162
GradientBoosting	0.339069697	0.189461347	0.884203577	0.343839302	0.194423291	0.885047271	122.6173909	0.09157753
DecisionTree	0.328243211	0.190337976	0.891480273	0.363318537	0.203162362	0.871653696	2.389936924	0.008504629
Ridge	0.364918777	0.199591699	0.865875039	0.357651994	0.201150449	0.875626013	0.062554598	0.001997709
Lasso	0.388163898	0.231950193	0.848243441	0.380902846	0.233802748	0.858929343	0.162287235	0.001999855
ElasticNet	0.37764959	0.216505236	0.856353442	0.368473638	0.217022077	0.867985663	0.14254117	0.002003193
BayesianRidge	0.364918775	0.199590762	0.86587504	0.357652082	0.201149689	0.875625952	0.155242682	0.002000809
PolynomialRegression	0.351536344	0.194920347	0.875532017	0.348868988	0.20064577	0.881659613	5.463344336	0.280403614
SGD	0.365614913	0.201223756	0.865362825	0.358816348	0.203073406	0.874814883	0.49190712	0.00200057
NeuralNetwork	0.340405384	0.193849549	0.883289473	0.3433571	0.200226014	0.885369465	70.20501637	0.019024372

Figure 8: Testing Result for Each Model (PM10)[10]

The comprehensive analysis of machine learning models for PM2.5 and PM10 prediction showcases a balance between accuracy and computational time. The Random Forest model emerges as the leading model due to its high R^2 score, indicating a strong ability to explain the variance in air quality data for both PM2.5 and PM10. While Neural Network offers the lowest RMSE for PM10, suggesting its proficiency in capturing fluctuations in pollutant levels, it trades off slightly in R^2 score.

Gradient Boosting stands out for PM2.5 with its blend of precision and efficiency, possessing a high R^2 and reasonable training time. On the other hand, simpler models like Ridge and Lasso, although quick, are outperformed by Random Forest and Gradient Boosting in terms of accuracy. Choosing the top three models involves considering not just the predictive accuracy but also the model's speed. For instance, while Decision Tree offers rapid predictions, it lacks the depth of accuracy seen in Random Forest and Gradient Boosting. The results indicate that more complex models, despite requiring longer computation, provide more accurate forecasts.

When assessing predictive performance as the primary criterion, Random Forest, Gradient Boosting, and Neural Network emerge as the top models due to their superior accuracy in forecasting PM2.5 and PM10 levels. However, it is important to note that other models also demonstrate commendable performance. The choice between these models should be informed by a careful consideration of the trade-off between predictive accuracy and computational efficiency. Users should weigh their specific requirements and constraints—whether prioritizing speed or precision—to select the most suitable model for their particular application.

[9] For visual insights and detailed figures related to the "Prediction of Air Quality in Beijing" study, please visit the GitHub repository. You can find the relevant illustrations and graphs in the *Figures_for_paper* folder.
[10] Same as foot note 8

In practical applications where time is a constraint, the speed of obtaining a reasonably accurate prediction may be prioritized. Conversely, when the highest level of accuracy is needed, and computational resources permit, more complex models would be preferred. Future directions in air quality forecasting should take into account these trade-offs to tailor the model selection to the specific demands of the task at hand.

Discussion

This study embarks on a comprehensive exploration of machine learning techniques to predict air quality levels in Beijing, focusing on PM2.5 and PM10 concentrations. Through careful preprocessing and optimization, the research identifies Random Forest, Gradient Boosting, and Neural Networks as top-performing models. These models demonstrated superior predictive accuracy, evidenced by low RMSE and MAE and high R^2 scores, indicating their potential for reliable and generalizable air quality forecasts.

The study also highlights the trade-off between model complexity and computational time. While more complex models such as Random Forest and Gradient Boosting offer heightened accuracy, they require more processing time compared to the efficient Neural Networks. This finding underlines the importance of tailored model selection, considering both accuracy and efficiency, to meet the specific demands of air quality forecasting. The research contributes to the growing body of knowledge on environmental data analysis and provides actionable insights for enhancing public health responses to air pollution challenges.

Appendix A: Team Member Contributions

Tianyi Zhang's Contributions: Tianyi Zhang played a crucial role in the coding aspects of the project. He was responsible for the data preprocessing and the selection of the appropriate models, a task that involved intricate coding and careful analysis. His expertise was instrumental in printing out the final results, ensuring accuracy and reliability. In the presentation phase, Tianyi took charge of developing the background information and explaining the methodology used in the study. His ability to articulate complex processes in an understandable manner was a key asset to the team.

Andy Han's Contributions: Andy Han's contributions were equally vital to the success of the project. In the coding phase, Andy's primary focus was on examining the raw data. He skillfully created graphs that visually represented this data, providing insightful and clear interpretations of the findings. During the presentation, Andy was responsible for crafting the motivation behind the work, giving a quick yet comprehensive overview, and analyzing the results. His ability to convey the significance and implications of the findings was central to the impact of the presentation.

Overall Team Dynamics: Both Tianyi Zhang and Andy Han contributed significantly and equally to the project. Their distinct but complementary skills combined to cover all aspects of the work, from detailed data handling to clear and effective communication of the findings. This collaboration highlights the strength of the team, where diverse expertise and shared commitment led to the successful completion of the project.

Appendix B: Set of Hyperparameters Evaluated in Grid Search

This appendix outlines the specific machine learning models employed in the study and the corresponding hyperparameters that are considered in the grid search process. The grid search aims to optimize these hyperparameters to enhance the performance of each model.

Models and Their Hyperparameters Grid:

1) ***Random Forest Regressor***
   - n_estimators: [50, 100, 150, 200]
   - max_depth: [None, 10, 20, 30]
   - min_samples_split: [2, 4, 6]
   - min_samples_leaf: [1, 2, 4]
2) ***Gradient Boosting Regressor***
   - n_estimators: [100, 200, 300, 400]
   - learning_rate: [0.01, 0.05, 0.1, 0.2]
   - max_depth: [3, 5, 7, 9]
3) ***Decision Tree Regressor***
   - max_depth: [None, 10, 20, 30, 40]
   - min_samples_split: [2, 4, 6, 8]
   - min_samples_leaf: [1, 2, 4]
4) ***Ridge Regression***
   - alpha: [0.1, 1, 10, 100]
5) ***Lasso Regression***
   - alpha: [0.1, 1, 10, 100]
6) ***ElasticNet Regression***
   - alpha: [0.1, 1, 10, 100]
   - l1_ratio: [0.2, 0.5, 0.7, 0.9]
7) ***Bayesian Ridge Regression***
   - alpha_1: [1e-6, 1e-5, 1e-4]
   - alpha_2: [1e-6, 1e-5, 1e-4]
8) ***Polynomial Regression (Pipeline)***
   - poly__degree: [2, 3]
9) ***SGDRegressor***
   - penalty: ['l2', 'l1', 'elasticnet']
   - max_iter: [1000, 2000]
   - tol: [1e-3, 1e-4]
10) ***Neural Network (MLP Regressor)***
   - hidden_layer_sizes: [(50,), (100,), (50, 50), (100, 100)]
   - max_iter: [500, 1000]

Appendix C: GitHub Repository

This GitHub repository contains resources for a study on air pollution in Beijing, focusing on PM2.5 and PM10 predictions using data from March 2013 to February 2017. It includes:

- Data Folder with raw, preprocessed, and split data.
- Scripts Folder with Python scripts for data processing, model optimization, and analysis.
- Usage instructions for setting up, preparing data, model selection, training, and analysis.
- List of required Python libraries.

Link to the GitHub repository: https://github.com/yrqoeuqo123/Prediction_of_Air_Quality_in_Beijing.git

Reference

AirNow.gov, U.S. EPA. (n.d.). Using Air Quality index. Using Air Quality Index | AirNow.gov. https://www.airnow.gov/aqi/aqi-basics/using-air-quality-index/#:~:text=AQI%20Forecasts&text=They%20use%20a%20number%20of,forecast%20for%20the%20next%20day.

California Air Resources Board. Inhalable Particulate Matter and Health (PM2.5 and PM10) | California Air Resources Board. (n.d.). https://ww2.arb.ca.gov/resources/inhalable-particulate-matter-and-health

Chen,Song. (2019). Beijing Multi-Site Air-Quality Data. UCI Machine Learning Repository. https://doi.org/10.24432/C5RK5G.chinapower2017. (2021, February 26). Is air quality in China a social problem?. ChinaPower Project. https://chinapower.csis.org/air-quality/

He, R. R. (2023). Quantifying the weekly cycle effect of air pollution in cities of China. Stochastic Environmental Research and Risk Assessment, 37, 2445–2457. https://doi.org/10.1007/s00477-023-02399-z

Hua, J., Zhang, Y., de Foy, B., Mei, X., Shang, J., & Feng, C. (2021). Competing PM2.5 and NO2 holiday effects in the Beijing area vary locally due to differences in residential coal burning and traffic patterns. The Science of the total environment, 750, 141575. https://doi.org/10.1016/j.scitotenv.2020.141575

Lyu, Y., Ju, Q., Lv, F., Feng, J., Pang, X., & Li, X. (2022). Spatiotemporal variations of air pollutants and ozone prediction using machine learning algorithms in the Beijing–Tianjin–Hebei region from 2014 to 2021. Environmental Pollution, 306, Article 119420.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer,

P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (1970, January 1). Scikit-Learn: Machine learning in Python. Journal of Machine Learning Research. https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html

Waskom, M. L., (2021). seaborn: statistical data visualization. Journal of Open Source Software, 6(60), 3021, https://doi.org/10.21105/joss.03021

World Health Organization. (1970, January 1). Who Global Air Quality Guidelines: Particulate matter (PM2.5 and PM10), ozone, nitrogen dioxide, sulfur dioxide and carbon monoxide. World Health Organization. https://apps.who.int/iris/handle/10665/345329

World Health Organization. (n.d.). Who Global Air Quality Guidelines: Particulate matter (PM2.5 and PM10), ozone, nitrogen dioxide, sulfur dioxide and carbon monoxide. World Health Organization. https://www.who.int/publications/i/item/9789240034228

Zhu, Y. Y., Wang, X. F., Wang, W., Dao, X., Wang, S., & Chen, S. R. (2022). Huan jing ke xue= Huanjing kexue, 43(3), 1212–1225. https://doi.org/10.13227/j.hjkx.202104329