Prediction of Air Pollution Concentration Using Weather Data and Regression Models

Aleksandar Trenchevski, Marija Kalendar, Hristijan Gjoreski and Danijela Efnusheva
Ss. Cyril and Methodius University, Faculty of Electrical Engineering and Information Technologies,
Rugjer Boshkovik 18, PO Box 574, 1000 Skopje, R.N. Macedonia
atrenchevski@gmail.com, {marijaka, hristijang, danijela}@feit.ukim.edu.mk

Keywords: Air Pollution, Feature Selection, Machine Learning, Prediction, Regression Models.

Abstract: Air pollution is becoming a global environmental problem, in both developed and developing countries. It has greatly impacted the health and lives of millions of people, thus increasing mortality rates and pollution induced diseases reports. This paper proposes machine learning methods for predicting the rates of possibly increased air pollution in several areas, by processing the gathered data from multiple weather and air quality meter stations. The data has been gathered over a period of several years including air quality and pollution data and weather data including temperature, humidity and wind characteristics. The development process included feature extraction, feature selection for removing redundancy, and finally training multiple regression models and hyperparameter optimization. Pollutants and air quality index (AQI) were used as target variables, and appropriate regression models were trained. The performed experiments show that XGBoost is the most accurate, achieving MAE of 8.9 for Center, 8.9 for Karpos and 7.3 for Kumanovo municipality for the PM10 pollutant. The improvements over the baseline, Dummy regressor are significant, reducing the MAE for 12 on average.

1 INTRODUCTION

The continuing increase in computing power and the development of many machine learning methods, currently opens up the possibility for massive data processing. One quite interesting and very particular research area of interest among scientists around the world encompasses climate changes and atmospheric impact by human behavior. Consequently, a quite vast amount of data describing atmospheric characteristics is being collected over a number of previous years. The existing data and the novel ways of data processing using machine learning algorithms, enable the scientists to gather, process and connect the data, and subsequently produce a novel view, relations and deductions. These new methods make it possible to detect the interconnections within the data, to present these results effectively, as well as to make some predictions based on the previous data occurrences.

Due to increasing demand for energy, population growth, economic development, urbanization and transportation, the problem of air pollution becomes the focus of modern society, primarily because of its adverse effects on human health, the environment and the climate system. It can be noted that the concentration of harmful substances in the atmosphere is constantly increasing, but this problem has only recently been approached with greater care. The main pollutants in the air are carbon oxides (carbon monoxide and carbon dioxide), nitrogen and sulphur oxides, particulate matter (PM2.5 and PM10), ammonia, some toxic metals, volatile organic compounds, etc.

Realizing that mere air monitoring means just scraping the surface of this enormous problem, the next step which is a challenge for scientists is the possible prediction of increased air pollution rates at particular time periods. This information could possibly aid the human population for health preservation, as well as governmental organizations responsible for controlling traffic and industrial capacities that have been identified as main polluters and source of the toxic materials present in the air.

Taking into account the seriousness of the researched area, this paper focuses on predicting the hourly air pollution for multiple locations across the country for the year 2018 using vast data merged in a dataset created from data gathered in the previous four years (2015 - 2018). Multiple regression models were used for processing the data in order to benchmark and pinpoint the most precise model that could be further developed and improved for the designated cause.

The rest of the paper is organized as follows: Section 2 presents related work in similar areas of research. Section 3 layouts the data gathering and preparation process. The used methodology has been described in Section 4. Section 5 presents the experimental results, and finally Section 6 concludes the paper.

2 RELATED WORKS

As a global environmental problem, air pollution has become a highly researched topic. Most researchers focus on monitoring and predicting the air quality index (AQI). As presented in [11] the prediction of air pollutants is extremely important for early warning and control of environmental pollution. Thus, developing models and forecasting the AQI (PM2.5 concentration) is very important to enable prevention and control of air pollution [12].

As a result, air pollution has been deeply researched all around the world and a vast variety of predictive models have been proposed. Researchers in Brazil [13] used a multilayer neural network for predicting hourly concentration of PM2.5 in Santiago, and identified the small dataset as reason for the poor predictions. Other researchers in China, [17], were evaluating hybrid regression models, EMD–SVR hybrid and EMD–IMF hybrid, achieving at most 80% accuracy, using only past AQI data to predict present AQI, not taking into consideration other correlation between different pollutants. Researchers in Italy [14] used feed-forward neural networks to predict ozone and PM10 in Milan. The predictions showed a satisfactory reliability, but the model still has the tendency toward overfitting. Another work, [15], uses recursive neural network model to forecast PM10 concentration for the next few days. The model showed 95% accuracy in predictions, but simultaneously yielding 30% false positives, which shows the limitations of neural networks models.

Another predictive model, the supplementary leaky integrator echo state network (SLI-ESN) is presented in [3]. This model aims to accurately predict the PM2.5 time series and, thus, implements different techniques to incorporate the historical information from the data and to consider the redundancy and correlation between multivariable time series. In order to achieve this, a minimum redundancy maximum relevance (mRMR) feature selection method is being introduced to reduce redundant and irrelevant information. The proposed model has been verified by experimenting with Beijing PM2.5 time series prediction. The experiments in [3] present the validity of the SLI-ESN model, showing high prediction accuracy in medium- and long-term projects, good generalization performance and good application prospects. Nevertheless, long-term predictions in [3] are not satisfactory and need to be improved.

The study presented in [4] focuses on using two regression algorithms, SVR and RFR, to build prediction models for the AQI in Beijing and the nitrogen oxides (NO_X) concentration in an Italian city based on publicly available datasets. The root-mean-square error (RMSE), correlation coefficient (r), and coefficient of determination (R2) were used to evaluate the performance of the regression models. Both models present good experimental results, but the complexity of the SVR model increased drastically with the increase of samples.

Focusing on our vicinity, [18] presents some results regarding developing multiple regression models for predicting the pollution mostly in suburban areas in Skopje. The models used vast number of features, subsequently reduced to the most important ones. Three approaches for building a model have been considered: single regression approach, ensemble approach and TPOT. Results showed that the ensemble-based methods (XGBoost) present quite good characteristics. Another conclusion is that PM10 appeared to be generally less predictable than the PM2.5 particles. Nevertheless, the obtained results from the used datasets were moderate, due to the incompleteness of the data, with major gaps of missing data.

3 DATA PREPARATION

Firstly, datasets publicly available at the Git repository of AirCare application [19], [20], were used in this research. The archived data of four years (2015 – 2018) gathered from air pollution measuring stations across the country of Republic of N. Macedonia has been used. Next, weather data gathered from the open API provided by DarkSky project [21], for cities and municipalities across the world was used, and in this case, data regarding

Republic of N. Macedonia. Both datasets have been combined into multiple reports, presenting merged information from pollutants as well as atmospheric details of the local weather.

A total of 10 weather/pollutant stations across the country were included in the research, and all of them had different measurement inconsistencies and huge time gaps that had to be conditioned when merging the data.

All of the available information about pollutants from the datasets were included in each of the consecutive steps: feature generation, feature selection, training and prediction phases, since they all contribute to the pollution rates and overall air quality.

Data conditioning included filtering out redundant and unnecessary data in the combined reports, since several variables had very few valid data values, depending on the station that monitored those values. Finally, the best decision was to eliminate such values. For other variables, filling out missing values had to be undertaken, using interpolation methods. This was justified for variables having missing values in very short time intervals, thus filling such missing information would not change the realistic values drastically and would not have great influence on the training phase. Some of the variables, like the air quality index (AQI), can be calculated as the maximum index of all pollutants. The last step included removal of potential outliers, finally completing the dataset to be ready for the next phase.

4 METHODOLOGY

4.1 Feature Selection

Feature selection is the process of selecting a subset of relevant features to use in the model construction. Appropriate feature selection enables accuracy improvement, overfitting risk reduction, speed up in training, improved data visualization, and increases the possibility for model understanding.

Time series data, affecting air pollution, contains rich, but also irrelevant and redundant information. This information reduces the accuracy of the predictions and efficiency of the model. Many feature selection algorithms exist, and they are distinguished by the evaluation metric into three main categories: filters, wrappers and embedded methods.

For this research, the backward elimination method from the wrapper category has been used, due to its precision for selecting relevant features based on the given machine learning model. Nevertheless, for a large number of features, the time complexity rises.

Backward Stepwise (Backward Elimination) Regression is a stepwise regression approach that begins with a full (saturated) model and at each step gradually eliminates variables from the regression model to find a reduced model that best explains the data. The stepwise approach is useful because it reduces the number of predictors, reducing the multi co-linearity problem and is one of the ways to resolve the overfitting.

The subset of features is generated separately for each station, target variable and machine learning model accordingly, in order to achieve maximum efficiency of the algorithm and better testing results.

4.2 Regression Learning

The selected subset of features is used as an input in the training of six regression models for predicting pollution values. The data for the year 2018 has been chosen to be used as the test dataset for each model.

The execution process starts from the first model and collects all the prediction values, errors from predicting, as well as the selected features for each iteration. There is need for some manual feature generation in order to add features derived from the timestamp and the previous value of the target variable, as well as categorical features which were needed to be hard-coded because the regression algorithm cannot process string object features.

Selecting proper parameters for tuning the efficiency of the model is calculated using randomized grid search due to the time complexity of the grid search algorithm for a large number of parameters.

4.2.1 Decision Tree Regression

The first regression model is the decision tree, which builds regression models in the form of a tree structure. It breaks down the dataset into smaller and smaller subsets, while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node has two or more branches, each representing values for the attribute tested. Leaf node represents a decision on the numerical target. The topmost decision node in a tree which corresponds to the best predictor is called a root node. Decision trees can handle both categorical and numerical data. This model would prove to be one of the most accurate models, ranking mostly at second or third place.

4.2.2 Dummy Regression

The dummy regression model is a baseline model, since it calculates the predictions by following a set of simple rules. It can be set to predict a fixed value calculated as the mean, median and quantile of the training set or as a constant given by the user. It was used as a baseline model to compare the rest of the models.

4.2.3 Light GBM Regression

The third, light GBM regression model, is a fast, distributed, high-performance gradient boosting framework, based on the decision tree algorithm, used for ranking, classification, regression and many other machine learning tasks. Its most significant characteristic is splitting the tree leaf-wise, and not depth-wise or level-wise, as other algorithms do. This enables better, faster and more accurate reduction decisions. The downside of leaf-wise splitting is increase in complexity and possible overfitting, overcome by specifying max-depth parameter where the splitting ends. Another feature of Light GBM, leading to faster training and higher efficiency is the histogram-based algorithm that buckets continuous feature values into discrete bins, thus also resulting in lower memory usage. Light GBM is suitable for use with large datasets, where it presents significant reduction in training time as compared to XGBoost.

4.2.4 Linear Regression

Linear regression is a machine learning algorithm based on supervised learning. The regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between the variables and the forecasting values. The relationship between dependent and independent variables is one of the main differentiators between regression models, as is the number of independent variables being used.

This regression technique finds out a linear relationship between x (independent variable, input) and y (dependent variable, output). Training the linear regression model means trying to find out coefficients for the linear function that best describe the input variables.

While building a linear model, the main goal is to minimize the error made by the algorithm while making predictions, which is done by choosing a function to help measure the error also called a cost function. The cost function, that help measure the error of the linear regression is the Root Mean Squared Error (RMSE) between the predicted y value and the true y value.

4.2.5 Random Forest Regression

Random Forest is a flexible, easy to use machine learning algorithm that produces great results most of the time with minimum time spent on hyper-parameter tuning. It has gained popularity due to its simplicity and the fact that it can be used for a great amount of regression tasks.

Random Forest is an ensemble machine learning technique capable of performing regression tasks using multiple decision trees and a statistical technique called bagging.

This algorithm builds multiple decision trees and merges their predictions together to get a more accurate and stable prediction rather than relying on individual decision trees.

The advantages of this model include: reduction in overfitting, its easy to measure the relative importance of each feature on the prediction, and it has an in-built validation mechanism named Out-of-bag.

However, the Random Forest model's disadvantages include: more complex and computationally expensive, slow and ineffective for real-time predictions, cannot extrapolate at all to data that is outside the range that the algorithm has seen.

Thus, Random Forest is a technique of many simple ideas combined together to yield an extremely accurate model.

4.2.6 Support Vector Regression

Support Vector regression (SVR) is characterized by the use of kernels, sparse solution and VC control of the margin and the number of support vectors. Although less popular than Support Vector Machine (SVM), SVR has been proven to be an effective tool in real-value function estimation. As a supervised-learning approach, SVR trains using a symmetrical loss function, which equally penalizes high and low misestimates.

One of the main advantages of SVR is that its computational complexity does not depend on the dimensionality of the input space. Additionally, it has excellent generalization capability, with high prediction accuracy.

4.2.7 XGBoost regression

XGBoost is an optimized distributed gradient boosting model designed to be highly efficient, flexible and portable. It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solves many data science problems in a fast and accurate way. Gradient Boosting Machines fit into a category of machine learning called Ensemble Learning, which is a branch of machine learning methods that train and predict with many models at once to produce a single superior output.

Ensemble learning is broken up into three primary subsets:

- Bagging: Bootstrap Aggregation or Bagging presents two features defining its training and prediction. For training, it uses a Bootstrap procedure to separate the training data into different random subsamples, which different iterations of the model use to train on. For prediction, a bagging regression takes an average of all models to produce output.
- Stacking: A Stacking model is a "meta-model" which uses the outputs from a collection of many, different models as input features. The idea is that this can reduce overfitting and improve accuracy.
- Boosting: The core definition of boosting is a method that converts weak learners to strong learners and is typically applied to trees. More explicitly, a boosting algorithm adds iterations of the model sequentially, adjusting the weights of the weak learners along the way. This reduces bias from the model and typically improves accuracy.

  Bagging along with boosting are two of the most popular ensemble techniques which aim to deal with high variance and high bias.

In conclusion, the XGBoost algorithm is optimized for modern data science problems and tools, it is highly scalable/parallelizable, quick to execute and typically outperforms other algorithms.

5 EXPERIMENTAL RESULTS

This section shows the comparison of the 7 regression algorithms. The dataset was split into 2 parts, 75% of the data for training (including the first three years, 2015 – 2017) and 25% for testing - i.e., year 2018. A cross-validation algorithm known as Randomized Search was used to determine the maximum potential of each algorithm.

The evaluation metrics used for ranking each of the results was Mean Absolute Error (MAE) as one of the most often used metrics with regression models. In MAE the error is calculated as an average of absolute differences between the target values and the predictions. MAE is a linear score, meaning that all individual differences are weighted equally in the average.

Figure 1 shows the comparison of MAE for each of the 7 algorithms for the PM10 pollutant in Centar municipality. The figure shows that the XGBoost is the most accurate algorithm for predicting the PM10 pollutant with a MAE value of 8.9. Light GBM is the second-best algorithm with a MAE of 9.1 and Random Forest with 10.4. The XGBoost performance is significantly better compared to the baseline - Dummy regressor.

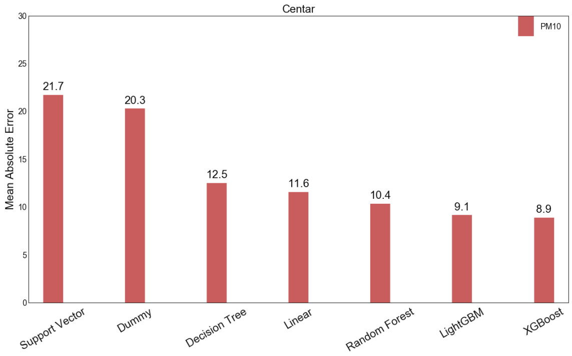

Figure 1: Mean Absolute Error plots for each algorithm predicting the PM10 pollutant in the center of Skopje.

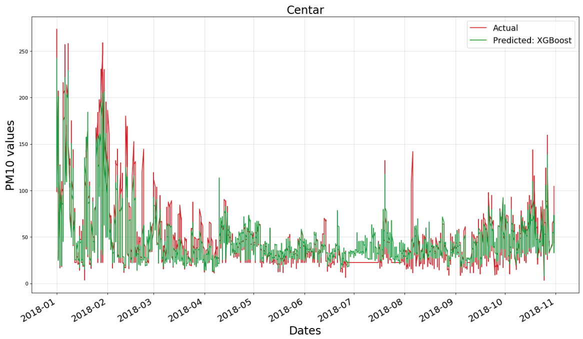

Figure 2: Actual and predicted values of the PM10 pollutant for each hour in 2018 in the center of Skopje.

Figure 2 shows the measured and the predicted values for the PM10 of the XGBoost algorithm. The values are shown for the period of 11 months in 2018. It can be noted that the predictions nicely follow the actual measurements.

Figure 3 shows the comparison of MAE for each of the 7 algorithms for the PM10 pollutant in Karpos municipality. The figure shows that XGBoost and LightGBM are the best performing, with a MAE value of 8.9 and 9.3 respectively, which is again a solid performance improvement by around 60% compared to the baseline - Dummy regressor.

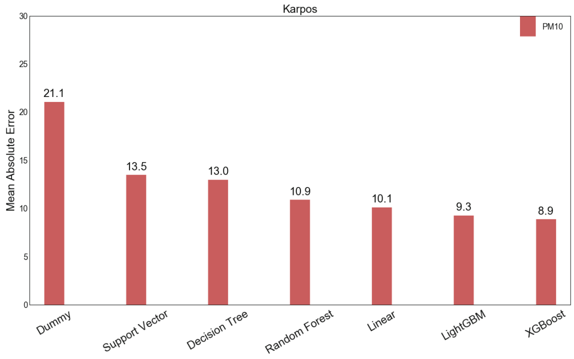

Figure 3: Mean Absolute Error for each algorithm predicting PM10 pollutant in.Karpos municipality - Skopje.

Figure 4 represents the comparison between the actual pollution data and the predictions from the XGBoost algorithm for Karpos. It shows that in general the predictions follow the actual measurements, and that there are underestimations in the predictions for the 10th and 11th month.

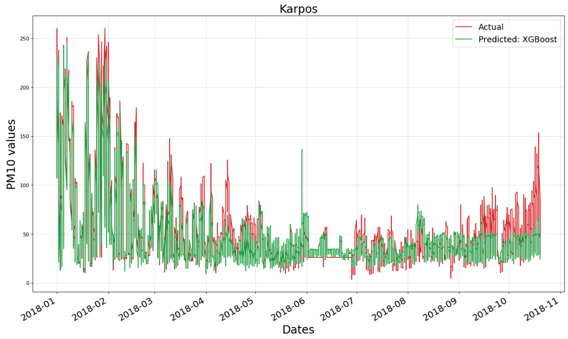

Figure 4: Actual and predicted values of PM10 pollutant for each hour in 2018 in the municipality of Karpos - Skopje.

Figure 5 shows the comparison of MAE for each of the 7 algorithms for the PM10 pollutant in Kumanovo municipality. Again, XGBoost and LightGBM are the best performing, with a MAE value of 7.3 and 8.0, respectively. This again results in a 60% performance improvement over the baseline algorithm.

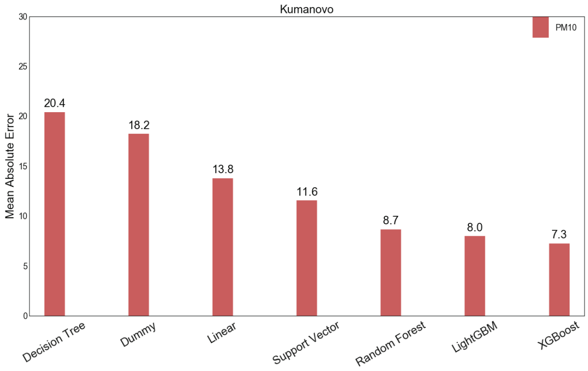

Figure 5: Mean Absolute Error for each algorithm predicting the PM10 pollutant in Kumanovo.

Figure 6 represents the comparison between the actual pollution data and the predictions from the XGBoost algorithm for Kumanovo. It shows that in general the predictions follow the actual measurements, and that there are underestimations in the predictions for the 5th month and the summer period.

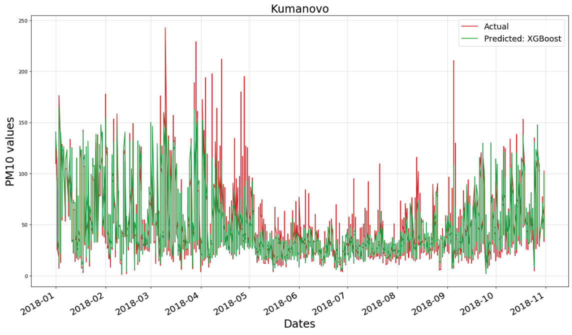

Figure 6: Actual and predicted values of the PM10 pollutant for each hour in 2018 in Kumanovo.

To summarize, the results show that XGBoost is the best performing algorithm, achieving MAE of 8.9 for Center, 8.9 for Karpos and 7.3 for Kumanovo. The improvements over the baseline, Dummy regressor are significant, reducing the MAE for 12 on average.

6 CONCLUSIONS

The paper presented a machine learning approach to predicting air pollution concentration, in particular PM10 concentration. The method uses the weather information and the previous pollution as an input, in order to calculate features and predict the PM10 concentration.

The first, and quite important step, is preparing and filtering the dataset so it can be ready for training and testing. This process eliminates unnecessary features, removes outliers that corrupt data and removes any inconsistencies with the target variables in order to preserve data integrity. The next step includes manual generation of useful features from already existing features in the dataset, using popular feature selection algorithms to improve overall dataset accuracy, as well as, using different cross-validation algorithms to achieve best results and obtain useful hyper parameters for each regression model. Finally, choosing evaluation metrics for dealing with prediction results from multiple regression models is necessary.

The overall results presented better performance than the baseline algorithm (Dummy Regression) by 60% and deliver a low mean absolute error which confirms the necessity of each mentioned step.

The incomplete data in the datasets played a major role in making the whole process harder to develop due to its inconsistency, a great deal of outliers, missing values that needed to be filled by interpolation or removed entirely. From around 30000 – 40000 rows of data it had to be cut down to around 15000 – 20000, which significantly lowers the accuracy of the model when training with half of the entire data.

REFERENCES

[1] Awad M., Khanna R., Support Vector Regression. In: Efficient Learning Machines. Apress, Berkeley, CA, 2015.

[2] Tuysuzoglu, G.; Birant, D.; Pala, A. Majority Voting Based Multi-Task Clustering of Air Quality Monitoring Network in Turkey. Appl. Sci., vol. 9, 2019, p. 1610.

[3] Xu, X.; Ren, W. Prediction of Air Pollution Concentration Based on mRMR and Echo State Network, Appl. Sci., vol. 9, 2019, p.1811.

[4] H. Liu, Q. Li, D. Yu, Yu Gu, Air Quality Index and Air Pollutant Concentration Prediction Based on Machine Learning Algorithms, Appl. Sci., vol. 9, 2019, p. 4069; doi:10.3390/app9194069.

[5] Backward Stepwise Regression. [Online] Available: http://www.analystsoft.com/en/products/statplus/content/help/analysis_regression_backward_stepwise_elimination_regression_model.html (28.12.2019).

[6] Decision Trees in Python with Scikit-Learn. [Online] Available: https://stackabuse.com/decision-trees-in-python-with-scikit-learn/ (28.12.2019).

[7] Linear Regression using Python. [Online] Available: https://medium.com/analytics-vidhya/linear-regression-using-python-ce21aa90ade6? (28.12.2019).

[8] Random Forest Regression model explained in depth. [Online] Available: https://gdcoder.com/random-forest-regressor-explained-in-depth/ (28.12.2019).

[9] Support Vector Regression Or SVR. [Online] Available: https://medium.com/coinmonks/support-vector-regression-or-svr-8eb3acf6d0ff (28.12.2019).

[10] A Step by Step Regression Tree Example. [Online] Available: https://sefiks.com/2018/08/28/a-step-by-step-regression-decision-tree-example/ (28.12.2019).

[11] S. Cai, Y. Wang, B. Zhao, S. Wang, X. Chang and J. Hao, "The impact of the "air pollution prevention and control action plan" on PM2.5 concentrations in Jing-Jin-Ji region during 2012-2020. Sci. Total Environ. 2017, 580, pp.197–209.

[12] L. Li, J.H. Zhang, W.Y. Qiu, J. Wang and Y. Fang, An Ensemble Spatiotemporal Model for Predicting PM2.5Concentrations. Int. J. Environ. Res. Public Health, vol. 14, 2017, p. 549.

[13] P. Pérez, A. Trier and J. Reyes, Prediction of PM2.5 concentrations several hours in advance using neural networks in Santiago, Chile. Atmos. Environ. 2000, 34, pp.1189-1196.

[14] G. Corani, Air quality prediction in Milan: Feed-forward neural networks, pruned neural networks and lazy learning. Ecol. Model. 2005, 185, 513–529.

[15] F. Biancofiore, M. Busilacchio, M. Verdecchia, B. Tomassetti, E. Aruo, S. Bianco, S. Di Tommaso, C. Colangeli, G. Rosatelli and P. Di Carlo, Recursive neural network model for analysis and forecast of PM10 and PM2.5. Atmos. Pollut. Res. 2017, 8, pp.652-659.

[16] G.W. Fuller, D.C. Carslaw and H.W. Lodge, "An empirical approach for the prediction of daily mean PM10 concentrations". Atmos. Environ. 2002, 36, pp.1431-1441.

[17] S. Zhu, X. Lian, H. Liu, J. Hu, Y. Wang, and J. Che, "Daily air quality index forecasting with hybrid models", A case in China. Environ. Pollut. 2017, 231, pp.1232-1244.

[18] P. Ilijevski, Gj. Smilevski, Predicting Air Pollution in Skopje, Project work for the course Data Warehouses and Data Processing.

[19] AirCare, Air Quality Visualized. [Online] Available: https://getaircare.com/.

[20] Pollution measurement data dumps, AirCare, December 2019. [Online] Available: https://github.com/jovanovski/MojVozduhExports.

[21] Dark Sky API, Weather Data on the Web., December 2019. [Online] Available: https://darksky.net/dev.