Title: An exploration of machine learning and statistical models for imputation, data augmentation, and prediction of indoor airborne pollutants concentrations

Using machine learning and statistical models to impute, augment, and predict indoor pollutant data from low-cost sensors

Ahmad Mohammadshirazi, George H. Fang, Ashvini A Kulshrestha, Jordan Clark, Rajiv Ramnath
mohammadshirazi.2@osu.edu, fang.855@osu.edu, kulshrestha.18@osu.edu, clark.1217@osu.edu, ramnath.6@osu.edu,

Abstract

Minimizing building energy consumption needed for maintenance of air quality in buildings can be aided by reliable predictions of future indoor airborne pollutant concentrations. In order to do this, this work pursues three objectives: (1) determining whether historical data from low-cost airborne pollutant sensors in buildings can be used to predict future indoor pollutant concentrations; 2) analyzing which algorithms and parameters are optimal for making these predictions,and (3) assessing the length of time into the future concentrations can be reliably predicted. Seven different methods (Rolling Average, Simple Linear Regression, Support Vector Machine, Random Forest, Gradient Boosting, Gated Recurrent Unit, and Long-Short Term Memory) are used to predict eight indoor pollutant concentrations (Carbon Dioxide, Nitrogen Dioxide, Ozone, three sizes of particulate matter, Formaldehyde, and Total Volatile Organic Compounds) in a single indoor location in California, and compared. Long-Short Term Memory was consistently the best method for predicting indoor pollutants (e.g. CO2, 1 hour ahead: MSE = 0.00231, 2 hours ahead: MSE = 0.00497, 3 hours ahead: MSE = 0.008454), though the best input combinations differed depending on the prediction forecasting time.

KEYWORDS: Data Driven, Indoor pollutants, Machine Learning, Prediction model regression, Prediction model classification, Gradient boosting, Random forest, Long Term Short Memory

1. Introduction

Emissions associated with energy used to operate buildings are a significant contributor to global warming[5.1]. In order to minimize energy consumption while at the same time providing for healthy indoor environments, a paradigm for control that includes sensing airborne pollutants of interest and predicting future concentrations may be of benefit.

1.1. Previous works

Prediction of variables associated with indoor air quality can prove challenging for several reasons: among them are the time needed for computation and the difficulty in performing accurate measurements. Feng et al., compared several simulation models and methods (CFD with coarse-grid, zonal, multizone, SFD, advanced turbulence, FFD, POD, Markov chain, TASIC, LLVM, RO-LLVM). The fast models themselves were not quick enough to reach real-time simulation for complex building designs; however, in combination with machine learning, the authors noted that it could be possible to satisfy the real-time or faster-than-real-time simulation requirements [6]. A review of 37 publications by Wei et al. demonstrated that linear regression was still comparable to artificial neural networks [6.1]. In a study by Lagasse et al., a collection of both statistical (multiple linear regression, partial least squares regression, distributed lag model, least absolute shrinkage selector operator) and machine learning models (simple artificial neural networks, long-short term memory) were used for prediction of particle concentrations and compared.. Though the study only modeled outdoor air pollution, machine learning still outperformed statistical models in predicting the concentration of $PM_{2.5}$ [8]. Liu et al.'s work used novel methods to boost the prediction accuracy of their LSTM model. The authors showed that by adding factory distribution features to increase awareness of concentrated outdoor pollution areas, the accuracy of the LSTM model in predicting outdoor $PM_{2.5}$ increased [8.1]. In addition, Zhang et al. developed a hybrid deep learning model VMDBiLSTM that combines variational mode decomposition (VMD) and bidirectional long short-term memory network (BiLSTM), to predict $PM_{2.5}$ changes in cities in China [5.2]. Li et al. also used the random forest machine learning model to predict indoor $PM_{2.5}$[9].In a study performed by Sharma et al., researchers developed two models to predict air quality characteristics. Using Multi-Layer Perceptron and eXtream Gradient Boosting Regression techniques, the estimation algorithm could estimate indoor air quality with at least 95% accuracy. [7]

1.2. Research Objectives

In this work, several machine learning and statistical models are applied to estimate eight indoor airborne pollutants: Carbon Dioxide (CO2), Nitrogen Dioxide (NO2), Ozone (O3), Particulate Matter 1 (PM1), Particulate Matter 2.5 (PM2.5), Particulate Matter 10 (PM10), Formaldehyde (CH2O), and Total Volatile Organic Compound (TVOC). The objective of this work is to explore the application of indoor air quality prediction methods using low-cost airborne pollutant sensors through a case study of a single commercial building in California. Specific objectives include

1. Determining whether historical data from low-cost airborne pollutant sensors in buildings can be used to predict future pollutant concentrations,
2. Analyzing which algorithms and parameters are optimal for making these predictions.
3. Assess the length of time into the future concentrations can be reliably predicted

2. Data

This work accomplishes three main tasks: imputing missing data, predicting future pollutant levels, and forecasting future peak concentrations. These tasks are part of an iterative research process, and each step builds upon the methods and results of the previous method.

Researchers from the Lawrence Berkeley National Laboratory in California collected data from a building in Alameda County. From October 25, 2019 to March 14, 2020 (dates inclusive), sensors located in in the building collected several datastreams including Carbon Dioxide (CO_2), Nitrogen Dioxide (NO_2), Ozone (O_3), Particulate Matter 1 (PM_1), Particulate Matter 2.5 ($PM_{2.5}$), Particulate Matter 10 (PM_{10}), Formaldehyde (CH_2O), and Total Volatile Organic Compound (TVOC) , Temperature (T), Humidity (RH), and Pressure (P) every minute. Additionally, time data (hour, the day of the week, the day of year) was used as inputs in prediction algorithms. The day of the week was represented by a numerical identifier (e.g. Monday is "1," Tuesday is "2," etc.), as well as the day of the year ("1" through "365").

Hourly outdoor data in Alameda County for the same set of pollutants and outdoor thermal variables were obtained from Berkeley Environmental Air-quality & CO2 Network, Air Now Tech, and Purple Air's publicly available data for Berkeley [16-18]. In order to normalize the conflicting per-minute indoor and hourly outdoor data, the indoor dataset was averaged to hourly values.

2.1 Imputation of missing data

Several data points from the hourly outdoor datasets were missing, and thus had to be imputed. Since only six data points were missing from the PM_1, $PM_{2.5}$, and PM_{10},datasets and only five data points were missing from the RH and T datasets (<0.20% of each total dataset), the previous data point was used to fill the missing records. For the CO_2, NO_2, and O_3 datasets, however, a significant number of data points was missing (CO2 = 251 points/7.61%, NO2 = 195 points/5.91%, O3 = 239 points/7.24%), so more complex methods were necessary to impute the missing data.

For the CO2, NO2, and O3 datasets, the RA, RF, GB, and LSTM methods were simultaneously applied to determine the best method of imputing the missing outdoor data. For the RA method, the average of the previous 7 data points within the dataset were used to fill the missing data points. For the RF method, 320 estimators were used when generating the trees. For the GB method, 100 estimators were used when generating the trees. For the LSTM-imputation method, to ensure accuracy, 1000 epochs, a batch size of 128, and a sub signal size of 7 were applied to the model. The activation function used was the sigmoid function.

2.1.1 Cross validation

To validate the imputed data, cross validation and comparison of the data's statistics were used. For each method, cross validation was used to test the new dataset on the model after it has already been trained. Five percent of the total training data was used for cross-validation, and the MSEs found from cross-validating each method were compared, except for averaging [18.4].

However, cross validation varied slightly in LSTM compared to all other methods. In LSTM, cross validation checks all pollutants at once, thus outputting three variables. This is in contrast to Rolling Average, Random Forest, and Gradient Boosting because all these methods tested pollutants individually.

To validate the imputed data, statistics from the data generated by each method are compared to the statistics of the original dataset. The statistics compared include average, mode, standard deviation, and median. The datasets were normalized before the statistics were computed. The method that has the most similar statistics to the original dataset is the most accurate method. The statistics of the dataset generated by each method can be found within the results section of the paper.

Results from imputing the missing data with four different methods- Rolling Average with 7 data points, Random Forest, Gradient Boosting, and LSTM- are shown below. Cross validation was applied here to allow for better comparison of the methods. Rolling aAverage was the least accurate method with MSEs for training loss ranging from 0.013 and 0.033. The Rolling Average method does not allow for a validation split, so no MSE for validation loss was calculated for this method.

With Random Forest, results on the training set exceeded that of all other methods, including LSTM. Training MSEs ranged between 0.001 and 0.006. However, results on the validation set indicated that the method did not perform as well with the new data. Validation MSEs had higher values, ranging between 0.003 and 0.015. This indicates some degree of overfitting for this technique, as it performs excellently on the data it was trained on, with less useful results on other previously unseen data.

Gradient Boosting was less accurate than Random Forest with training MSEs between 0.004 and 0.015. The validation MSEs ranged between 0.006 and 0.02.

LSTM had comparable results to Random Forest. Training MSEs from imputing data using LSTM ranged between 0.001 and 0.007. Validation MSEs ranged from 0.001 to 0.003. This indicates that the LSTM methods did not experience overfitting as Random Forest did, while still retaining a high degree of accuracy, making it the method of choice. Table 3 displays all the training and validation MSEs obtained for CO2, NO2 and O3.

Cross Validation Method	CO_2_out	NO_2_out	O_3_out
Average	$MSE_{T.L.}$: 0.013625 $MSE_{V.L.}$: N/A	$MSE_{T.L.}$: 0.021807 $MSE_{V.L.}$: N/A	$MSE_{T.L.}$: 0.032612 $MSE_{V.L.}$: N/A
Random Forest	$MSE_{T.L.}$: 0.000575 $MSE_{V.L.}$: 0.003734	$MSE_{T.L.}$: 0.00172 $MSE_{V.L.}$: 0.012139	$MSE_{T.L.}$: 0.00137 $MSE_{V.L.}$: 0.014704
Gradient Boosting	$MSE_{T.L.}$: 0.00453 $MSE_{V.L.}$: 0.006918	$MSE_{T.L.}$: 0.00933 $MSE_{V.L.}$: 0.011387	$MSE_{T.L.}$: 0.01511 $MSE_{V.L.}$: 0.019305
LSTM	**$MSE_{T.L.}$: 0.0016** **$MSE_{V.L.}$: 0.0014**	**$MSE_{T.L.}$: 0.00611** **$MSE_{V.L.}$: 0.0023**	**$MSE_{T.L.}$: 0.00605** **$MSE_{V.L.}$: 0.0023**

Table 3. MSEs during training and validation of outdoor CO_2, NO_2, and O_3.

3. Predicting data points

Using the fully imputed dataset, the next goal was to find accurate predictions among eight indoor pollutants, including Carbon Dioxide (CO2), Nitrogen Dioxide (NO2), Ozone (O3), and Particulate Matter (PM1, PM2.5, and PM10). This work utilises seven different methods for pollutant prediction: Rolling Average (RA)[0], Simple Linear Regression (LR)[1], Support Vector Machine (SVM)[2], Random Forest (RF)[3], Gradient Boosting (GB)[4], Gated Recurrent Unit[4.1], and Long-Short Term Memory (LSTM)[5]. To prevent statistical bias, four different ratios of testing-training (5%, 10%, 15%, and 20%) and 10% validation data were chosen, the model was trained five times using each ratio, and then the results of the five iterations were averaged.

The inputs for the prediction models make use of indoor and outdoor pollutant datasets, as well as the temporal data (hour, day of the week, day of the year). Table 4 summarizes the original set of inputs used to determine the optimal set of inputs.

Predicted Pollutant	Inputs
Carbon Dioxide (CO2)	Hour, CO2_in, RH_in, T_in, P_in, CO2_out, RH_out, T_out, num_week
Nitrogen Dioxide (NO2)	Hour, NO2_in, RH_in, T_in, P_in, NO2_out, RH_out, T_out, num_week
Ozone (O3)	Hour, O3_in, RH_in, T_in, P_in, O3_out, RH_out, T_out, num_week
Particulate Matter (PM1)	Hour, PM1_in, RH_in, T_in, P_in, PM1_out, RH_out, T_out, num_week
Particulate Matter (PM2.5)	Hour, PM2.5_in, RH_in, T_in, P_in, PM2.5_out, RH_out, T_out, num_week
Particulate Matter (PM10)	Hour, PM10_in, RH_in, T_in, P_in, PM10_out, RH_out, T_out, num_week
Formaldehyde (CH2O)	Hour, CO2_in, RH_in, T_in, CH2O_in, T_out, num_year, num_week
Total Volatile Organic Compound (TVOC)	Hour, CO2_in, RH_in, T_in, TVOC_in, T_out, num_year, num_week

Table 4: Complete set of possible inputs for each pollutant

To reduce the complexity and training time of each model, a combinatory pattern recognition model is developed (feature selection), and smaller combinations of the input values were used as opposed to all 20 variables. This process was also shown to reduce MSE when compared to models trained with all 20 input variables. For each pollutant, there are 1023 (2^10 - 1) possible combinations due to ten variables (the indoor pollutant being tested for, the outdoor pollutant being tested for, indoor time, indoor pressure, outdoor relative humidity, day in the year, day in the week, and hour) being considered. MSE was used as the benchmark to find the optimal combination, and adjusted R-Squared was also reported as a secondary confirmation metric, though was not considered until the final model was trained.

3.1 Results of predicting indoor pollutants

Using a variety of methods, we analyzed the prediction of indoor air quality using different relative timespans. Predicting pollutant levels from 1 hour ahead, 2 hours ahead, and 3 hours ahead yielded a wide range of results depending on the method (see Tables 5-7). Notably, for the gradient boosting and random forest models, a significant difference existed between the training and testing MSE's; the models were overfit to the training data, causing inaccurate testing predictions. To predict 2-hours and 3-hours ahead using the LSTM-prediction model, lag was introduced (1 hour ahead = 1 lag, etc.).

Method	CO_2	NO_2	O_3	PM_1	$PM_{2.5}$	PM_{10}	CH_2O	TVOC
Average	[0.045697] -0.20% {0.037571} (-3.79%)	[0.040134] 25.40% {0.048218} (23.34%)	[0.028170] 53.64% {0.072395} (1.46%)	[0.011167] 75.98% {0.008050} (58.14%)	[0.012636] 74.43% {0.007295} (55.69%)	[0.012184] 73.55% {0.007983} (50.04%)	[0.010090] 34.36% {0.078506} (2.54%)	[0.004979] 29.27% {0.011034} (35.20%)
Simple Linear Regression	[0.004211] 84.60% {0.004074} (79.18%)	[0.002590] 86.78% {0.005571} (75.24%)	[0.001792] 93.08% {0.003640} (89.89%)	[0.002202] 94.08% {0.001652} (89.82%)	[0.001781] 92.93% {0.001090} (88.80%)	[0.001939] 92.59% {0.001296} (87.63%)	[0.002416] 75.16% {0.018251} (62.78%)	[0.002461] 73.74% {0.0665} (7.71%)
Support Vector Machine	[0.009413] 65.56% {0.007525} (61.54%)	[0.014461] 26.16% {0.013803} (38.64%)	[0.011778] 54.51% {0.010928} (69.64%)	[0.017985] 51.62% {0.015666} (3.53%)	[0.018422] 26.88% {0.017721} (-82.11%)	[0.020356] 22.21% {0.018079} (-72.52%)	[0.017119] -82.74% {0.065166} (-9.58%)	[0.018729] -92.61% {0.027808} (43.29%)
Random Forest	[0.000441] 98.27% {0.003197} (83.55%)	[0.000316] 98.27% {0.004543} (76.60%)	[0.000192] 99.23% {0.003554} (88.82%)	[0.000248] 99.31% {0.001611} (89.14%)	[0.000201] 99.17% {0.001096} (88.12%)	[0.000217] 99.13% {0.001271} (86.92%)	[0.000349] 96.27% {0.054821} (23.93%)	[0.000398] 95.90% {0.019477} (60.28%)
Gradient Boosting	[0.002277] 90.58% {0.003235} (83.42%)	[0.001446] 91.71% {0.004437} (77.50%)	[0.001172] 95.16% {0.003119} (90.03%)	[0.001102] 96.90% {0.001595} (89.52%)	[0.000787] 96.73% {0.001083} (88.31%)	[0.000803] 99.13% {0.001275} (86.97%)	[0.001467] 84.35% {0.055129} (23.51%)	[0.001015] 89.56% {0.019530} (60.17%)
GRU	[0.003088] 88.36% {0.002997} (78.37%)	[0.002744] 83.81% {0.003732} (79.85%)	[0.001918] 91.81% {0.002601} (88.11%)	[0.001981] 93.43% {0.001403} (84.67%)	[0.001571] 93.04% {0.000852} (85.23%)	[0.001611] 92.38% {0.001022} (83.51%)	[0.000190] 77.16% {0.004971} (13.89%)	[0.000871] 77.62% {0.004718} (63.55%)
LSTM	[0.002938] 88.93% {0.002310} (79.72%)	[0.002339] 88.61% {0.002421} (82.99%)	[0.001775] 94.14% {0.002339} (90.03%)	[0.001739] 94.89% {0.000777} (84.34%)	[0.001372] 93.99% {0.000448} (87.14%)	[0.001462] 93.83% {0.000632} (86.16%)	[0.000168] 76.45% {0.004415} (19.66%)	[0.000566] 78.08% {0.004515} (64.60%)

Table 5: Hour-ahead indoor pollutant prediction [Training Mean Squared errors] Training Adjusted R-Squared {Testing Mean Squared errors] (Testing Adjusted R-Squared)

Hour Prediction	CO_2	NO_2	O_3	PM_1	$PM_{2.5}$	PM_{10}	CH_2O	TVOC
1 Hour Ahead	**[0.002938] 88.93% {0.002310} (79.72%)**	**[0.002339] 88.61% {0.002421} (82.99%)**	**[0.001775] 94.14% {0.002339} (90.03%)**	**[0.001739] 94.89% {0.000777} (84.34%)**	**[0.001372] 93.99% {0.000448} (87.14%)**	**[0.001462] 93.83% {0.000632} (86.16%)**	**[0.000168] 76.45% {0.004415} (19.66%)**	**[0.000566] 78.08% {0.004515} (64.60%)**

2 Hour Ahead	**[0.004419] 83.34% {0.00497} (56.45%)**	**[0.003855] 81.23% {0.005916} (58.49%)**	**[0.00305] 89.93% {0.004614} (80.28%)**	**[0.003275] 90.37% {0.002884} (81.41%)**	**[0.002779] 88.92% {0.001777} (80.89%)**	**[0.002823] 89.15% {0.002084} (79.14%)**	**[0.004786] 49.81% {0.011470} (-18.08%)**	**[0.000566] 78.08% {0.004515} (64.60%)**
3 Hour Ahead	**[0.00838] 69.55% {0.008454} (56.33%)**	**[0.008380] 69.55% {0.008454} (56.33%)**	**[0.003791] 85.03% {0.01057} (70.88%)**	**[0.005723] 84.54% {0.004398} (71.68%)**	**[0.003358] 86.61% {0.002552} (72.57%)**	**[0.003504] 86.54% {0.002781} (72.18%)**	**[0.000168] 76.45% {0.004415} (19.66%)**	**[0.005474] 37.63% {0.04675} (2.40%)**

Table 6: Comparison of LSTM-prediction model performance for prediction intervals [Training Mean Squared errors] Training Adjusted R-Squared {Testing Mean Squared errors] (Testing Adjusted R-Squared)

Figure 1 compares the 6 methods used. LSTM is the most accurate method amongst all 6 that were applied.

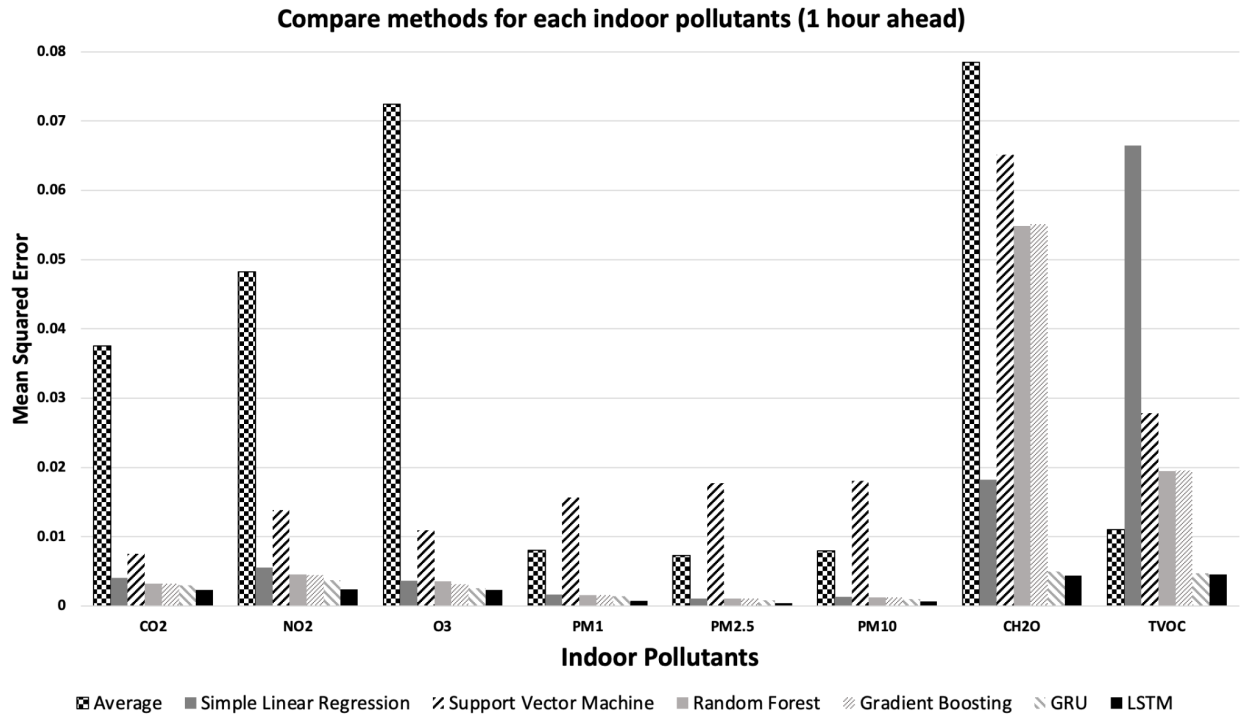
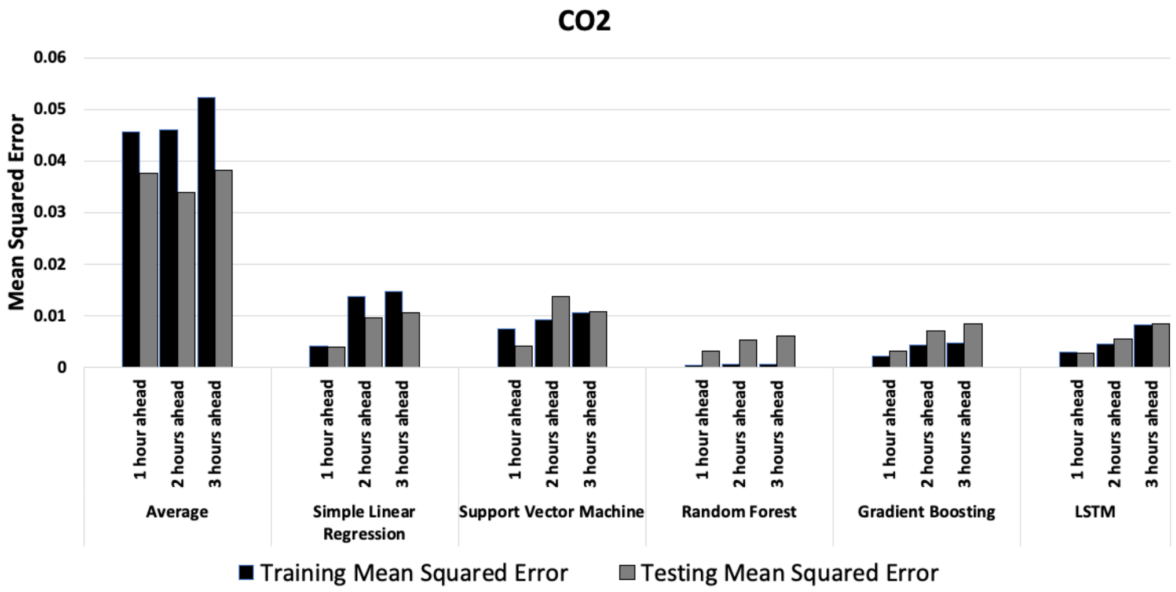
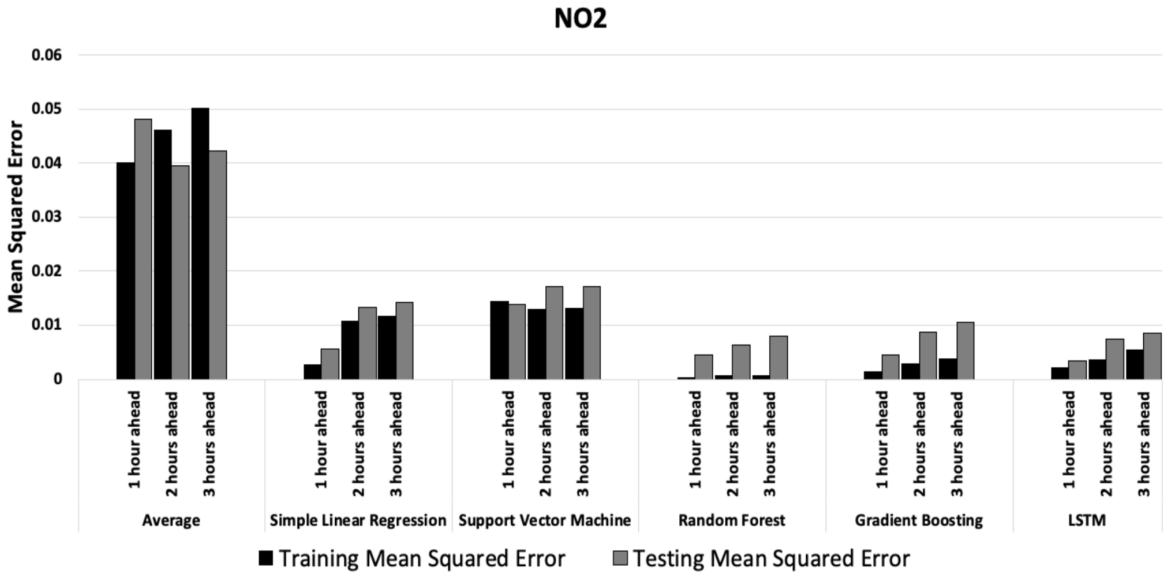
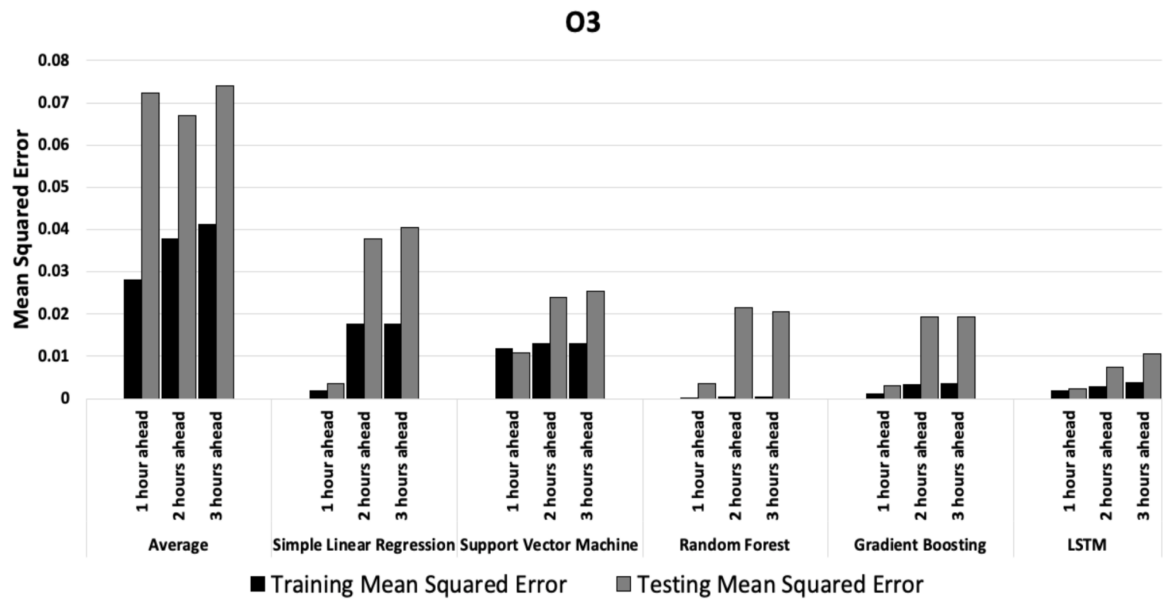


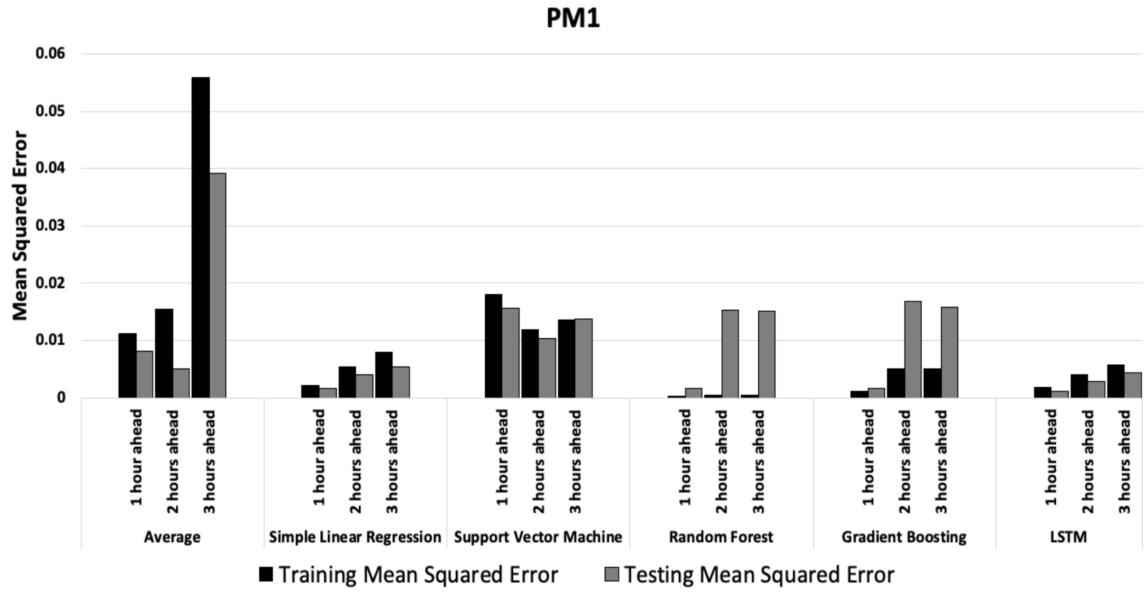
Figure 1: Comparison of methods for 1-hour prediction

Unlike gradient boosting and random forest models, LSTM was consistently the best model for predicting indoor pollutants (see Figure 1), though the best input combinations differed depending on the prediction interval (see Tables 5-7). For an hour-ahead prediction, testing MSEs for concentrations of CO2 were 0.002310, NO2 were 0.00241, O3 were 0.002339, PM1 were 0.000777, PM2.5 were 0.000448, PM10 were 0.000632, CH2O were 0.004415, and TVOC were 0.004515.


CO2
Mean Squared Error
0.06
0.05
0.04
0.03
0.02
0.01
0
1 hour ahead
2 hours ahead
3 hours ahead
Average
Simple Linear Regression
Support Vector Machine
Random Forest
Gradient Boosting
LSTM
Training Mean Squared Error
Testing Mean Squared Error


NO2
Mean Squared Error
0.06
0.05
0.04
0.03
0.02
0.01
0
1 hour ahead
2 hours ahead
3 hours ahead
Average
Simple Linear Regression
Support Vector Machine
Random Forest
Gradient Boosting
LSTM
Training Mean Squared Error
Testing Mean Squared Error


O3
Mean Squared Error
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
1 hour ahead
2 hours ahead
3 hours ahead
Average
Simple Linear Regression
Support Vector Machine
Random Forest
Gradient Boosting
LSTM
Training Mean Squared Error
Testing Mean Squared Error


PM1
Mean Squared Error
0.06
0.05
0.04
0.03
0.02
0.01
0
1 hour ahead
2 hours ahead
3 hours ahead
Average
Simple Linear Regression
Support Vector Machine
Random Forest
Gradient Boosting
LSTM
Training Mean Squared Error
Testing Mean Squared Error

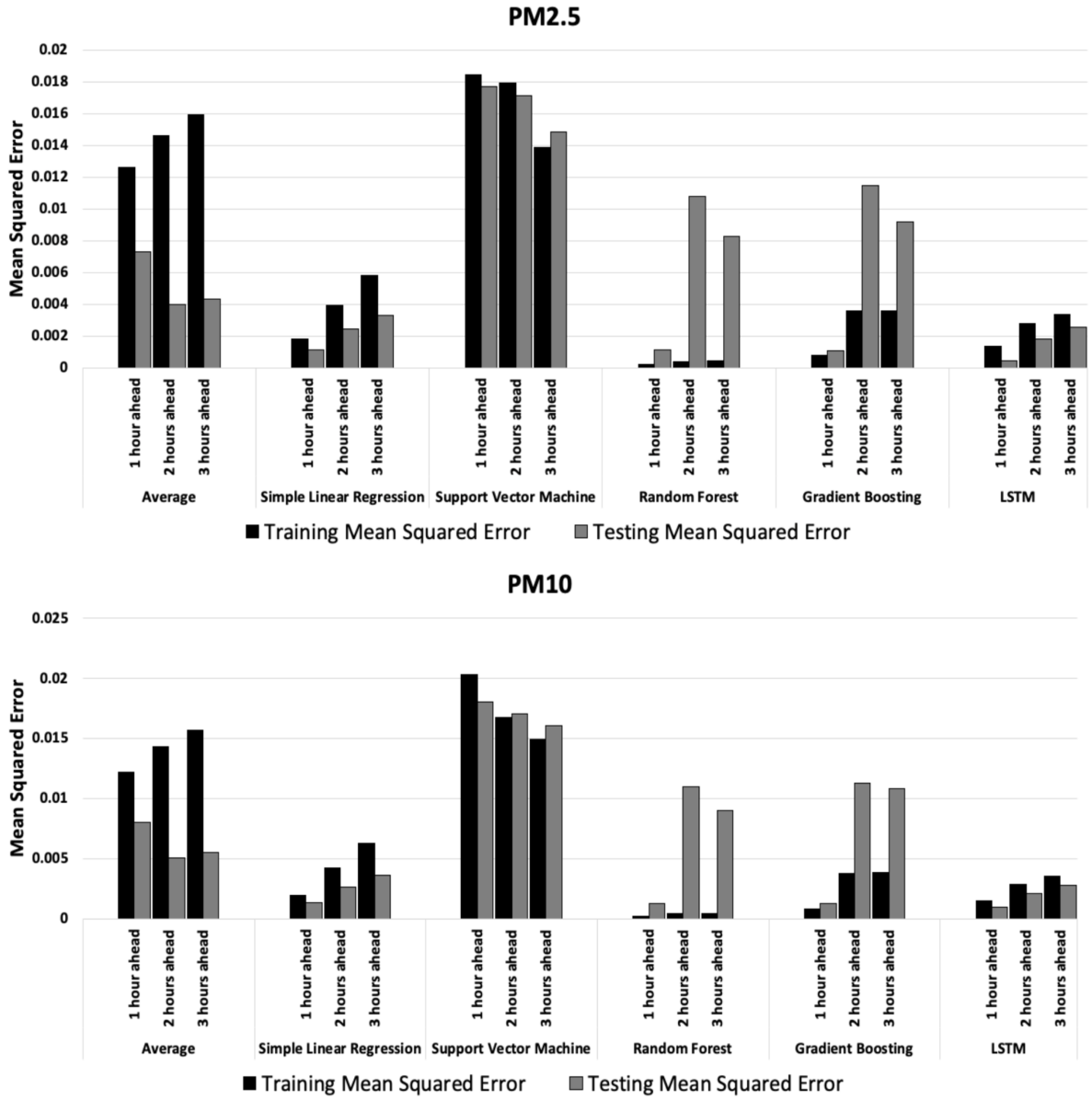


Figure 2: Comparison of methods for 1-3 hour prediction

Figure 2 displays a comparison of MSE values for the predicted time periods (1-3 hours ahead) based on the prediction method. The LSTM-prediction method performed consistently better with the least error, while the rolling average method performed consistently worse than the other methods. The gradient boosting and random forest models show significant overfitting to the training data.

3.2 Generate synthetic data:

It is hypothesized that given more data, a more accurate model can be produced. Thus, this work used a Generative Adversarial Network (GAN) to generate synthetic data to test this hypothesis. A GAN model using 40,000 epochs, 32 batch size, 24 hidden dimensions and 3 layers as calibrated hyperparameters (see Table 5) was trained to generate 3300 extra data points. Of those data points, 20 individual variables were generated. These individual variables were indoor variables, outdoor variables, and the hour of the data point. Indoor-specific variables include CO_2, NO_2, O_3, PM_1, $PM_{2.5}$, PM_{10}, RH, T, P, CH_2O, and TVOC. Outdoor variables include CO_2, NO_2, O_3, PM_1, $PM_{2.5}$, PM_{10}, RH, and T.

Since the dataset is sequential temporal data, a Gated Recurrent Unit (GRU) model and Long Term Short Memory (LSTM) model are applied. Principal Component Analysis (PCA) [10.1] and t-Distributed Stochastic Neighbor Embedding (t-SNE) [10.2] are used to measure the quality of the generated data and then reduce the results' dimensional complexity (since there are 20 hyperparameters, it is necessary to reduce the 20 dimensions created to 2 in order to display the data).

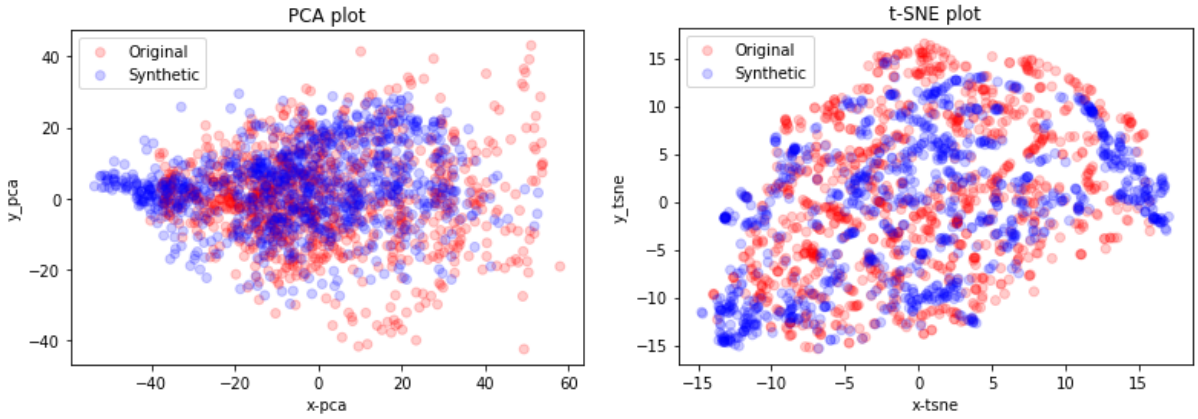


Figure 3: Original vs. Synthetic Data - Gated Recurrent Unit model (GRU)

The left plot displays Principal Component Analysis (PCA), and the right plot displays t-Distributed Stochastic Neighbor Embedding (t-SNE) for the GRU model. The predictive score is 72.81 percent.

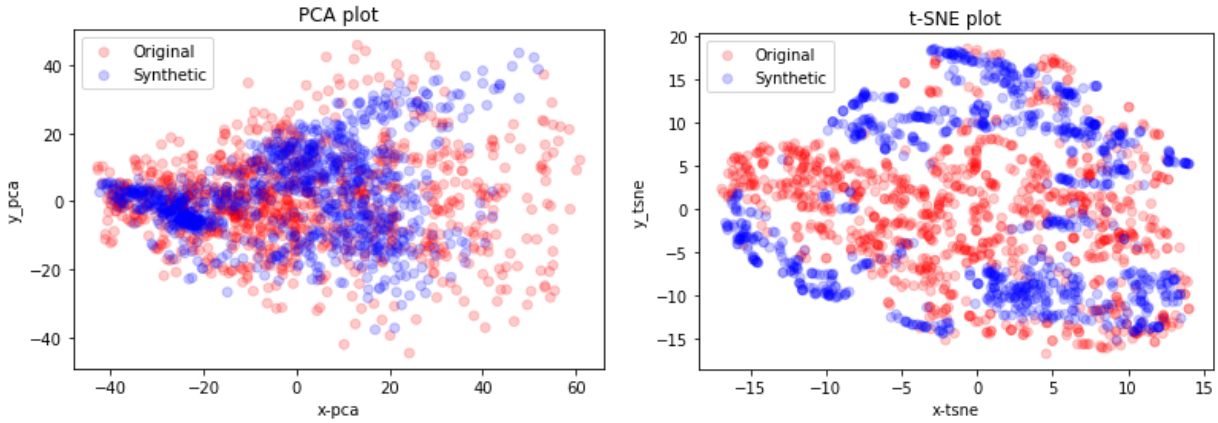


Figure 4: Original vs. Synthetic Data - Long-Term Short Memory (LSTM)

The left plot displays Principal Component Analysis (PCA), and the right plot displays t-Distributed Stochastic Neighbor Embedding (t-SNE) for the LSTM model. The predictive score is 68.99 percent.

Pollutant	Methods	Training MSE Improvement	Training R2 Improvement	Testing MSE Improvement	Testing R2 Improvement
CO2	LSTM	-0.000516	3.88%	-0.00032	5.76%
	GRU	0.00561	-26.04%	0.000659	0.11%
NO2	LSTM	-0.000338	3.30%	-0.000387	2.92%
	GRU	0.001115	-8.28%	0.000535	-1.62%
O3	LSTM	-0.000299	1.17%	-0.000465	3.62%
	GRU	0.000645	-4.58%	-0.0003	0.90%

PM1	LSTM	-0.00015	0.10%	-0.00012	4.21%
	GRU	0.00016	-1.51%	0.000298	0.56%
PM2.5	LSTM	-0.000101	0.82%	-0.000018	0.79%
	GRU	0.0001	-2.72%	0.000235	-2.37%
PM10	LSTM	-0.000175	1.20%	-0.000089	2.11%
	GRU	0.000371	-11.15%	0.000243	0.16%

Table 9: Comparison of original and synthetic-optimized LSTM-GAN and GRU-GAN models

After the synthetic data points were generated, they were appended to the training dataset, in effect doubling the total dataset. The testing data was not altered in order to retain the data's authenticity, and thus the RTT values of the model with synthetic data is half of that of the original model. The original model was then retrained using the entire 6600-value dataset to produce a synthetic-optimized model, which was more accurate than the original model in predicting indoor pollutant concentration levels (see Table 9).

4. Hyperparameter optimization

In order to create optimal machine learning models, it is necessary to configure certain hyperparameters that control the learning processes.

4.1 LSTM-Imputation

For the LSTM-imputation model, rigorous testing was used to find the optimal architecture and parameters. The best model consisted of a neural network with 3 hidden layers: LSTM layer (12 neurons), dense layer (30 neurons), dense layer (15 neurons). Each layer was accompanied by a dropout layer with a 25% dropout rate to prevent overfitting. The final settings were 1000 epochs and a batch size of 128 using the Adam learning algorithm. The loss function used mean squared error. For the imputed data, the 1300 data points with no missing values were used as the training dataset, while the total 4300 data points were used for testing. Once the training for the imputation model was complete, the model was then used to impute any missing data points.

Hyperparameters	Description	Searched	Selected
$n^{[lstm]}$	Number of neurons in the LSTM layer	[8,10,12,14,16,18,20]	12
$l^{[lstm]}$	Number of Dense layers	[1,2,3,4]	2
R	Repetition	[1, 5]	5
k	Number of epochs	[100, 200, 500, 1000]	1000
Size_in	Input sub-signal size	[8-24]	7
V.V.	Cross Validation Value		0.05
Batch Size	Batch Size Value	[32, 64, 128, 256]	128
Dropout	Drop Out	[0.1, 0.15, 0.20, 0.25, 0.30]	0.25
loss	Loss function	MSE	MSE
opt	Optimization method		Adam

g()	Activation function		ReLU

4.2 LSTM-Prediction

The optimal LSTM model consisted of a neural network with a hidden LSTM layer (12 neurons) and a dropout layer with a 25% dropout rate to prevent overfitting. Hyperparameters (see Table 4) were fine-tuned to predict indoor pollutants. 1000 epochs, 256 batch size, Adam optimization algorithm, MSE loss function, 12-hour subsignal, and 10% validation set are notable hyperparameters.

Hyperparameters	Description	Searched	Selected
$n^{[lstm]}$	Number of neurons in the LSTM layer	[8,10,12,14,16,18,20]	12
$l^{[lstm]}$	Number of layers in the LSTM layer	[1,2,3,4]	1
R	Repetition	[1, 5]	5
C	Combinations		1023
RTT	Ratio of Testing-Training	[5%, 10%, 15%, 20%]	
k	Number of epochs	[100, 200, 500, 1000]	1000
Size_in	Input sub-signal size	[8-24]	12
Size_ou	Output sub-signal size	[1-3]	[1-3]
V.V.	Cross Validation Value	0.05 - 0.1	0.1
Batch Size	Batch Size Value	[32, 64, 128, 256]	256
Dropout	Drop Out	[0.1, 0.15, 0.20, 0.25, 0.30]	0.25
loss	Loss function	MSE, Adjusted R^2	MSE
opt	Optimization method		Adam
g()	Activation function		ReLU

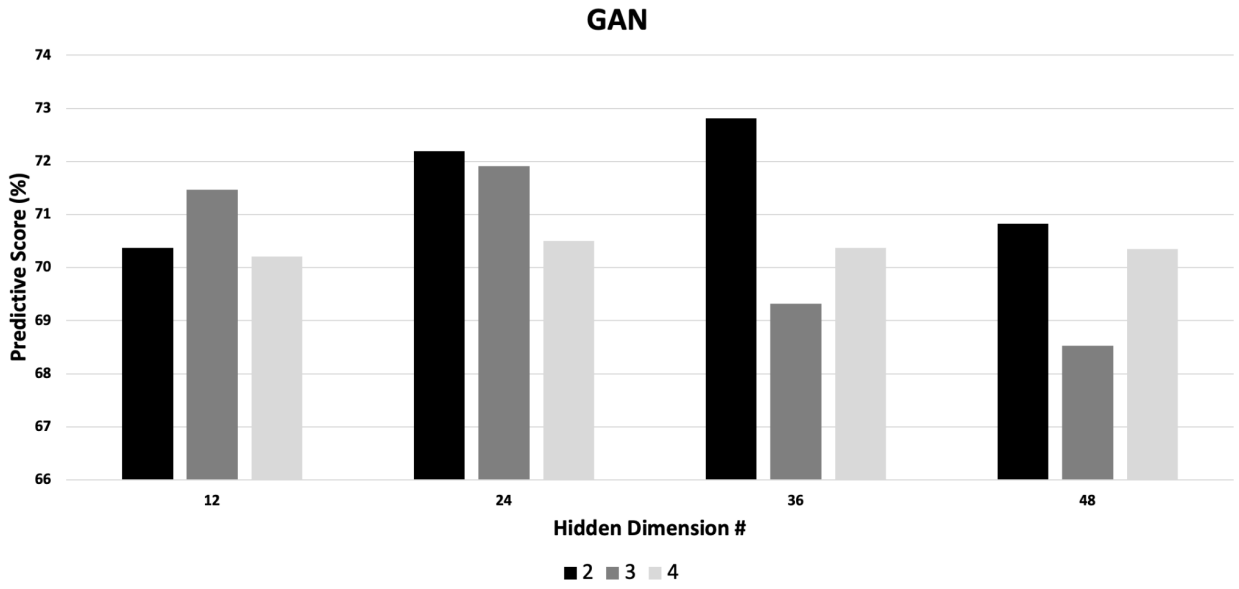
Table 4: Grid search hyperparameter tuning for LSTM-prediction

4.3 LSTM-GAN

For the LSTM-GAN model, a set of possible values were chosen for the number of hidden dimensions (hidden_dim), number of layers (l), number of epochs (iterations), batch size, and subsignal (sub_in) hyperparameters (see Table 8). Firstly, the batch size was set to the smallest value, as higher values decrease the model's accuracy. To determine the best combination of the remaining hyperparameters, all combinations of the number of hidden dimensions and number of layers were chosen (with 40,000 epochs and a 12-hour subsignal set as default values), and the resulting models were tested. Once 36 hidden dimensions and 2 layers were determined to be the best combination, each epoch value was then chosen and their models were tested. After determining 40,000 epochs was the optimal value, the same process was repeated for the subsignal hyperparameter. The best hyperparameter configuration was thus 36 hidden dimensions; 2 layers; 40,000 epochs; a batch size of 32, and a subsignal of 12 hours.

Hyperparameters	Description	Searched	Selected
hidden_dim	Number of Hidden Dimensions	[12, 24,36,48]	**36**
l[lstm-GRU]	Number of hidden layers	[2,3,4]	**2**
Iterations	Epochs	[1000, 10000, 15000, 25000, 40000, 50000]	**40,000**
Batch Size	Batch Size Value	[32, 64, 128, 256]	**32**
Sub_in	Subsignal	[8, 10, 12, 14]	**12**

Table 8: Hyperparameters for the Generative Adversarial Network (GAN) model



5. Sensitivity Analysis

Table 9 displays the results of CO_2 for the 5 best combinations of inputs for each of the 4 RTT values. The process was repeated exactly 5 times for each combination, and the results were averaged. (1) symbolizes the inclusion of the variable as part of the model, while (0) symbolizes its exclusion. The negative values represent the change in MSE due to the current state of the input as either included or excluded. For example, if the hour input were removed from the number 1 best combination using 5% RTT, it would increase the MSE by 0.00045. The two smallest changes in MSE are the inclusion of the number of days in a year in the number 2 and 3 combination using 10% RTT (-.00000). The two largest changes in MSE are the exclusion of indoor CO_2 from the number 4 combination using 15% RTT (-0.00674) and the exclusion of relative humidity from the number 5 combination using 15% RTT (-0.00628).

The last row of Table 9 sums how many times each variable has appeared in the top 5 combinations of the 4 train test split ratios for each estimation model, showing the statistical importance of each of the estimation variables in accurate model estimations. For example, for CO2 estimation, hour has appeared in all top 5 combinations of all train test split ratios (4 train test split ratio × 5 top combinations = 20).

CO2

RTT	Best Comb	Hour	CO2_in	RH_in	T_in	P_in	CO2_out	RH_out	T_out	num_year	num_week
	1	-0.00045 (1)	-0.00386 (1)	-0.00023 (0)	-0.00046 (1)	-0.00019 (1)	-0.00035 (1)	-0.00009 (1)	-0.00042 (1)	-0.00013 (1)	-0.00013 (1)
	2	-0.00036 (1)	-0.00321 (1)	-0.00017 (0)	-0.00034 (1)	-0.00009 (0)	-0.00017 (1)	-0.00004 (1)	-0.00010 (1)	-0.00015 (0)	-0.00007 (0)
5%	3	-0.00023 (1)	-0.00271 (1)	-0.00008 (0)	-0.00033 (1)	-0.00008 (0)	-0.00032 (1)	0.00004 (0)	-0.00024 (1)	-0.00013 (0)	-0.00014 (0)
	4	-0.00023 (1)	-0.00308 (1)	-0.00002 (0)	-0.00043 (1)	-0.00026 (1)	-0.00015 (1)	0.00009 (0)	-0.00036 (1)	-0.00018 (1)	-0.00011 (1)
	5	-0.00041 (1)	-0.00323 (1)	-0.00011 (1)	-0.00022 (1)	-0.00007 (1)	-0.00045 (1)	-0.00015 (0)	-0.00036 (1)	-0.00007 (1)	-0.00001 (0)
	1	-0.00043 (1)	-0.00378 (1)	-0.00005 (0)	-0.00020 (1)	-0.00001 (1)	-0.00014 (1)	-0.00004 (1)	-0.00016 (1)	-0.00004 (0)	-0.00005 (1)
	2	-0.00067 (1)	-0.00478 (1)	-0.00011 (0)	-0.00026 (1)	-0.00003 (0)	-0.00036 (1)	-0.00007 (1)	-0.00023 (1)	**-0.00000 (1)**	-0.00005 (1)
10%	3	-0.00044 (1)	-0.00393 (1)	-0.00010 (0)	-0.00033 (1)	0.00001 (0)	-0.00023 (1)	-0.00012 (1)	-0.00015 (1)	**0.00000 (0)**	-0.00005 (1)
	4	-0.00044 (1)	-0.00366 (1)	-0.00008 (0)	-0.00021 (1)	0.00003 (1)	-0.00005 (1)	-0.00006 (1)	-0.00019 (1)	0.00004 (1)	-0.00005 (1)
	5	-0.00041 (1)	-0.00389 (1)	-0.00002 (0)	-0.00017 (1)	-0.00008 (1)	-0.00004 (1)	0.00004 (0)	-0.00024 (1)	-0.00006 (0)	-0.00017 (1)
	1	-0.00074 (1)	-0.00441 (1)	-0.00017 (0)	-0.00027 (1)	-0.00016 (1)	-0.00034 (1)	-0.00008 (1)	-0.00019 (1)	-0.00009 (1)	-0.00027 (1)
	2	-0.00074 (1)	-0.00548 (1)	-0.00012 (1)	-0.00016 (1)	-0.00007 (0)	-0.00042 (1)	-0.00008 (1)	-0.00025 (1)	-0.00014 (0)	-0.00018 (1)
15%	3	-0.00071 (1)	-0.00439 (1)	-0.00007 (0)	-0.00009 (1)	-0.00006 (1)	-0.00029 (1)	-0.00005 (0)	-0.00003 (1)	-0.00005 (0)	-0.00008 (1)
	4	-0.00077 (1)	**-0.00674 (1)**	-0.00008 (0)	-0.00013 (1)	-0.00020 (1)	-0.00032 (1)	-0.00007 (0)	**0.00003 (0)**	-0.00013 (0)	-0.00018 (1)
	5	-0.00088 (1)	**-0.00628 (1)**	-0.00004 (0)	-0.00018 (1)	-0.00006 (1)	-0.00023 (1)	0.00008 (0)	-0.00011 (1)	0.00005 (1)	-0.00027 (1)
	1	**-0.00079 (1)**	**-0.00537 (1)**	**-0.00018 (0)**	**-0.00026 (1)**	**-0.00008 (0)**	**-0.00044 (1)**	**-0.00013 (1)**	**-0.00021 (1)**	**-0.00027 (0)**	**-0.00012 (1)**
	2	**-0.00057 (1)**	**-0.00488 (1)**	**-0.00008 (0)**	**-0.00017 (1)**	**-0.00008 (1)**	**-0.00023 (1)**	**-0.00002 (0)**	**-0.00024 (1)**	**-0.00008 (0)**	**-0.00007 (1)**
20%	3	-0.00054 (1)	-0.00582 (1)	-0.00006 (1)	-0.00011 (1)	-0.00006 (0)	-0.00033 (1)	-0.00011 (0)	-0.00011 (1)	-0.00017 (0)	-0.00012 (1)
	4	-0.00058 (1)	-0.00741 (1)	-0.00022 (1)	-0.00024 (1)	-0.00011 (1)	-0.00047 (1)	-0.00017 (0)	-0.00006 (0)	-0.00022 (0)	-0.00018 (1)
	5	-0.00049 (1)	-0.00501 (1)	-0.00004 (0)	-0.00008 (1)	0.00008 (1)	-0.00017 (1)	0.00002 (1)	-0.00011 (1)	-0.00005 (0)	-0.00002 (1)
		20	20	4	20	13	20	10	18	7	17

NO2

RTT	Best Comb	Hour	NO2_in	RH_in	T_in	P_in	NO2_out	RH_out	T_out	num_year	num_week
	1	-0.00014 (0)	-0.00532 (1)	-0.00006 (1)	-0.00012 (1)	-0.00008 (0)	-0.00013 (1)	-0.00010 (1)	-0.00009 (1)	-0.00005 (0)	-0.00003 (1)
	2	-0.00012 (0)	-0.00581 (1)	-0.00002 (1)	-0.00010 (1)	-0.00005 (0)	-0.00009 (1)	-0.00016 (1)	-0.00009 (1)	-0.00005 (0)	0.00003 (0)
5%	3	-0.00015 (0)	-0.00549 (1)	0.00002 (0)	-0.00011 (1)	-0.00007 (0)	-0.00011 (1)	-0.00010 (1)	-0.00007 (1)	-0.00009 (0)	-0.00001 (0)
	4	-0.00012 (0)	-0.00619 (1)	-0.00006 (1)	-0.00004 (1)	-0.00008 (0)	-0.00009 (1)	-0.00012 (1)	-0.00009 (1)	0.00005 (1)	-0.00003 (1)

	5	-0.00013 (0)	-0.00485 (1)	0.00006 (0)	-0.00012 (1)	-0.00002 (0)	-0.00009 (1)	-0.00006 (1)	-0.00010 (1)	-0.00006 (0)	0.00001 (1)
	1	-0.00008 (0)	-0.00485 (1)	-0.00009 (1)	-0.00014 (1)	-0.00010 (0)	-0.00014 (1)	-0.00011 (1)	-0.00010 (1)	-0.00004 (0)	-0.00006 (1)
	2	-0.00009 (0)	-0.00457 (1)	-0.00004 (0)	-0.00016 (1)	-0.00007 (0)	-0.00010 (1)	-0.00006 (1)	-0.00018 (1)	-0.00004 (1)	-0.00008 (0)
10%	3	-0.00005 (0)	-0.00522 (1)	-0.00007 (1)	-0.00001 (1)	-0.00013 (0)	-0.00012 (1)	-0.00011 (1)	-0.00018 (1)	0.00004 (1)	-0.00003 (1)
	4	-0.00001 (0)	-0.00599 (1)	-0.00015 (1)	0.00001 (0)	-0.00006 (0)	-0.00007 (1)	-0.00017 (1)	-0.00019 (1)	-0.00008 (1)	-0.00000 (1)
	5	-0.00005 (0)	-0.00476 (1)	-0.00013 (1)	-0.00001 (0)	-0.00004 (0)	-0.00010 (1)	-0.00021 (1)	-0.00014 (1)	-0.00001 (1)	0.00000 (0)
	1	-0.00017 (1)	-0.00634 (1)	-0.00004 (0)	-0.00003 (0)	-0.00004 (0)	-0.00050 (1)	-0.00011 (1)	-0.00013 (1)	-0.00018 (0)	-0.00014 (1)
	2	-0.00013 (1)	-0.00566 (1)	-0.00004 (1)	-0.00006 (0)	-0.00004 (1)	-0.00052 (1)	-0.00013 (1)	-0.00005 (1)	-0.00011 (0)	-0.00003 (1)
15%	3	-0.00026 (1)	-0.00503 (1)	-0.00001 (0)	-0.00009 (1)	-0.00011 (0)	-0.00053 (1)	-0.00001 (0)	-0.00029 (1)	-0.00027 (0)	-0.00010 (1)
	4	-0.00038 (1)	-0.00981 (1)	-0.00012 (1)	-0.00013 (0)	-0.00024 (1)	-0.00058 (1)	-0.00009 (0)	-0.00018 (1)	-0.00010 (1)	-0.00014 (1)
	5	-0.00041 (1)	-0.00522 (1)	-0.00008 (0)	-0.00006 (1)	-0.00005 (1)	-0.00064 (1)	-0.00016 (1)	-0.00003 (0)	-0.00026 (0)	-0.00022 (1)
	1	-0.00018 (1)	-0.00687 (1)	-0.00002 (1)	-0.00008 (0)	-0.00002 (1)	-0.00048 (1)	-0.00012 (1)	-0.00012 (1)	-0.00012 (0)	-0.00002 (0)
	2	-0.00009 (1)	-0.00670 (1)	-0.00001 (1)	-0.00001 (0)	-0.00005 (1)	-0.00052 (1)	-0.00002 (1)	-0.00006 (1)	-0.00004 (0)	0.00002 (1)
20%	3	-0.00014 (1)	-0.00559 (1)	-0.00010 (1)	-0.00011 (0)	0.00002 (0)	-0.00043 (1)	-0.00013 (1)	-0.00012 (1)	-0.00005 (0)	-0.00005 (0)
	4	-0.00014 (1)	-0.00731 (1)	-0.00001 (1)	-0.00005 (1)	-0.00001 (0)	-0.00052 (1)	-0.00004 (1)	-0.00006 (1)	-0.00019 (0)	-0.00011 (1)
	5	-0.00027 (1)	-0.00682 (1)	0.00002 (0)	-0.00012 (0)	-0.00010 (1)	-0.00052 (1)	-0.00008 (1)	-0.00013 (1)	-0.00016 (0)	-0.00001 (0)
		10	20	13	11	6	20	18	19	6	13

O3

RTT	Best Comb	Hour	O3_in	RH_in	T_in	P_in	O3_out	RH_out	T_out	num_year	num_week
	1	-0.00038 (1)	-0.07181 (1)	-0.00005 (0)	-0.00019 (1)	-0.00001 (0)	-0.00010 (1)	-0.00020 (0)	-0.00025 (0)	-0.00011 (1)	-0.00002 (1)
	2	-0.00038 (1)	-0.02943 (1)	-0.00012 (0)	-0.00011 (1)	0.00001 (1)	-0.00008 (1)	-0.00013 (0)	-0.00011 (0)	-0.00005 (1)	-0.00001 (1)
5%	3	-0.00035 (1)	-0.41655 (1)	-0.00006 (0)	-0.00011 (1)	-0.00000 (0)	-0.00002 (1)	-0.00012 (0)	-0.00020 (0)	-0.00006 (1)	0.00002 (0)
	4	-0.00035 (1)	-0.04018 (1)	-0.00011 (0)	-0.00006 (1)	0.00000 (1)	-0.00005 (1)	-0.00006 (0)	-0.00019 (0)	-0.00005 (1)	0.00001 (0)
	5	-0.00041 (1)	-0.05481 (1)	-0.00010 (0)	-0.00013 (1)	-0.00003 (0)	0.00002 (0)	-0.00007 (0)	-0.00018 (0)	-0.00003 (1)	-0.00006 (0)
	1	-0.00032 (1)	-0.01832 (1)	-0.00018 (0)	-0.00031 (1)	-0.00015 (1)	-0.00004 (1)	-0.00020 (0)	-0.00027 (0)	-0.00013 (1)	-0.00011 (1)
	2	-0.00022 (1)	-0.03440 (1)	-0.00010 (0)	-0.00020 (1)	-0.00002 (1)	0.00004 (0)	-0.00004 (0)	-0.00010 (0)	-0.00001 (1)	-0.00006 (1)
10%	3	-0.00032 (1)	-0.01572 (1)	-0.00012 (0)	-0.00026 (1)	-0.00011 (1)	-0.00008 (0)	-0.00006 (0)	-0.00014 (0)	0.00001 (0)	-0.00006 (1)
	4	-0.00022 (1)	-0.03751 (1)	-0.00001 (0)	-0.00023 (1)	0.00002 (0)	-0.00009 (0)	-0.00004 (0)	-0.00015 (0)	-0.00011 (1)	-0.00000 (1)
	5	-0.00029 (1)	-0.10540 (1)	-0.00005 (0)	-0.00021 (1)	-0.00004 (0)	-0.00008 (0)	-0.00002 (0)	-0.00016 (0)	-0.00005 (1)	0.00000 (0)
	1	-0.00037 (1)	-0.02268 (1)	-0.00015 (1)	-0.00028 (1)	-0.00021 (1)	-0.00025 (0)	-0.00045 (0)	-0.00019 (0)	-0.00025 (0)	-0.00017 (1)
15%											

	2	-0.00038 (1)	-0.03016 (1)	-0.00025 (0)	-0.00030 (1)	-0.00010 (0)	-0.00009 (1)	-0.00009 (1)	-0.00027 (0)	-0.00026 (0)	-0.00017 (1)
	3	-0.00022 (1)	-0.04096 (1)	-0.00012 (0)	-0.00045 (1)	-0.00008 (1)	-0.00008 (0)	-0.00008 (0)	-0.00026 (0)	-0.00015 (0)	-0.00010 (0)
	4	-0.00032 (1)	-0.03283 (1)	-0.00039 (0)	-0.00024 (1)	-0.00014 (1)	-0.00008 (1)	-0.00007 (1)	-0.00037 (0)	-0.00018 (0)	-0.00007 (0)
	5	-0.00031 (1)	-0.02296 (1)	-0.00016 (0)	-0.00033 (1)	-0.00009 (0)	-0.00005 (0)	-0.00005 (0)	-0.00021 (0)	-0.00015 (0)	-0.00008 (1)
	1	-0.00021 (1)	-0.05521 (1)	-0.00018 (0)	-0.00032 (1)	-0.00011 (1)	-0.00018 (0)	-0.00017 (0)	-0.00025 (0)	-0.00034 (0)	-0.00008 (0)
	2	-0.00031 (1)	-0.02911 (1)	-0.00027 (0)	-0.00020 (1)	-0.00006 (1)	-0.00023 (0)	-0.00008 (1)	-0.00028 (0)	-0.00044 (0)	-0.00016 (1)
20%	3	-0.00026 (1)	-0.04731 (1)	-0.00021 (0)	-0.00021 (1)	0.00006 (0)	-0.00020 (0)	-0.00003 (1)	-0.00007 (0)	-0.00061 (0)	-0.00000 (1)
	4	-0.00024 (1)	-0.04915 (1)	-0.00020 (0)	-0.00007 (1)	-0.00010 (0)	-0.00007 (0)	-0.00004 (1)	-0.00006 (0)	-0.00051 (0)	0.00000 (0)
	5	-0.00018 (1)	-0.03637 (1)	-0.00018 (0)	-0.00011 (1)	-0.00002 (1)	-0.00011 (0)	0.00008 (0)	-0.00007 (0)	-0.00039 (0)	0.00008 (1)
		20	20	1	20	11	7	5	0	9	12

pm1

RTT	Best Comb	Hour	PM1_in	RH_in	T_in	P_in	PM1_out	RH_out	T_out	num_year	num_week
	1	-0.00006 (1)	-0.00103 (1)	-0.00000 (0)	-0.00006 (1)	-0.00001 (0)	-0.00011 (1)	-0.00002 (0)	-0.00000 (0)	-0.00001 (0)	-0.00002 (1)
	2	-0.00001 (1)	-0.00087 (1)	-0.00001 (0)	-0.00005 (1)	-0.00001 (0)	-0.00010 (1)	-0.00000 (0)	0.00000 (1)	-0.00002 (0)	-0.00002 (1)
5%	3	-0.00002 (1)	-0.00096 (1)	-0.00001 (0)	-0.00007 (1)	-0.00002 (0)	-0.00008 (1)	0.00000 (1)	-0.00001 (1)	-0.00001 (0)	-0.00002 (1)
	4	-0.00006 (1)	-0.00108 (1)	0.00000 (1)	-0.00005 (1)	-0.00000 (0)	-0.00008 (1)	-0.00000 (0)	-0.00001 (0)	-0.00003 (0)	-0.00001 (1)
	5	-0.00007 (1)	-0.00106 (1)	-0.00000 (0)	-0.00005 (1)	0.00001 (1)	-0.00016 (1)	-0.00002 (0)	-0.00001 (0)	-0.00001 (0)	-0.00003 (1)
	1	-0.00004 (1)	-0.00091 (1)	-0.00001 (0)	-0.00008 (1)	-0.00002 (0)	-0.00017 (1)	-0.00005 (0)	-0.00001 (0)	-0.00004 (1)	-0.00002 (1)
	2	-0.00002 (1)	-0.00103 (1)	0.00001 (1)	-0.00007 (1)	-0.00002 (0)	-0.00017 (1)	-0.00002 (0)	-0.00000 (0)	-0.00003 (1)	-0.00002 (1)
10%	3	-0.00002 (1)	-0.00084 (1)	-0.00000 (1)	-0.00004 (1)	-0.00002 (0)	-0.00017 (1)	-0.00000 (0)	0.00000 (1)	-0.00005 (1)	-0.00004 (1)
	4	-0.00001 (1)	-0.00076 (1)	0.00000 (0)	-0.00003 (1)	-0.00002 (0)	-0.00017 (1)	-0.00001 (0)	0.00001 (1)	-0.00001 (1)	-0.00001 (1)
	5	-0.00000 (1)	-0.00080 (1)	-0.00001 (1)	-0.00005 (1)	-0.00004 (0)	-0.00017 (1)	0.00000 (1)	-0.00002 (1)	-0.00004 (1)	-0.00002 (1)
	1	-0.00009 (1)	-0.00369 (1)	-0.00003 (1)	-0.00013 (1)	-0.00005 (0)	-0.00015 (1)	-0.00004 (0)	-0.00003 (0)	-0.00011 (0)	-0.00003 (1)
	2	-0.00006 (1)	-0.00300 (1)	-0.00009 (0)	-0.00011 (1)	-0.00008 (1)	-0.00018 (1)	-0.00006 (0)	-0.00001 (1)	-0.00006 (1)	-0.00003 (1)
15%	3	-0.00004 (1)	-0.00398 (1)	-0.00003 (0)	-0.00020 (1)	-0.00003 (1)	-0.00011 (1)	-0.00003 (0)	0.00001 (0)	-0.00001 (1)	-0.00004 (1)
	4	-0.00005 (1)	-0.00582 (1)	-0.00003 (0)	-0.00014 (1)	-0.00001 (1)	-0.00016 (1)	-0.00002 (0)	-0.00004 (0)	0.00001 (0)	-0.00003 (1)
	5	-0.00001 (0)	-0.00597 (1)	-0.00002 (0)	-0.00007 (1)	-0.00003 (0)	-0.00018 (1)	-0.00003 (0)	-0.00003 (1)	-0.00005 (0)	-0.00004 (1)
	1	-0.00005 (1)	-0.00376 (1)	-0.00002 (1)	-0.00012 (1)	-0.00003 (0)	-0.00013 (1)	-0.00004 (0)	-0.00004 (0)	-0.00013 (0)	-0.00004 (1)
	2	-0.00002 (1)	-0.00379 (1)	0.00002 (0)	-0.00013 (1)	-0.00006 (0)	-0.00011 (1)	-0.00000 (0)	-0.00000 (0)	-0.00005 (0)	-0.00003 (1)
20%	3	-0.00004 (1)	-0.00415 (1)	-0.00002 (0)	-0.00009 (1)	-0.00003 (0)	-0.00012 (1)	0.00000 (1)	-0.00002 (0)	-0.00009 (0)	-0.00002 (1)

	4	-0.00002 (0)	-0.00413 (1)	-0.00002 (1)	-0.00006 (1)	-0.00001 (0)	-0.00014 (1)	-0.00002 (0)	-0.00003 (1)	-0.00011 (0)	-0.00003 (1)
	5	-0.00002 (1)	-0.00541 (1)	-0.00002 (0)	-0.00004 (1)	-0.00004 (0)	-0.00013 (1)	-0.00002 (0)	0.00000 (1)	-0.00005 (0)	-0.00003 (1)

18

pm2.5

RTT	Best Comb	Hour	PM2.5_in	RH_in	T_in	P_in	PM2.5_out	RH_out	T_out	num_year	num_week
	1	-0.00004 (1)	-0.00090 (1)	-0.00001 (1)	-0.00004 (1)	-0.00003 (0)	-0.00008 (1)	-0.00001 (0)	-0.00001 (0)	-0.00003 (0)	-0.00002 (1)
	2	-0.00003 (1)	-0.00074 (1)	-0.00002 (1)	-0.00004 (1)	-0.00003 (0)	-0.00008 (1)	-0.00001 (0)	-0.00002 (1)	-0.00001 (1)	-0.00002 (1)
5%	3	-0.00002 (1)	-0.00063 (1)	-0.00000 (0)	-0.00004 (1)	-0.00002 (0)	-0.00009 (1)	-0.00001 (1)	-0.00002 (1)	-0.00000 (1)	-0.00003 (1)
	4	-0.00000 (0)	-0.00074 (1)	-0.00001 (1)	-0.00003 (1)	-0.00003 (0)	-0.00008 (1)	-0.00002 (0)	-0.00003 (1)	-0.00003 (0)	-0.00002 (1)
	5	-0.00001 (1)	-0.00069 (1)	-0.00003 (0)	-0.00002 (1)	-0.00001 (0)	-0.00007 (1)	-0.00000 (1)	-0.00000 (1)	0.00000 (0)	-0.00001 (1)
	1	-0.00004 (1)	-0.00065 (1)	-0.00002 (0)	-0.00005 (1)	-0.00002 (0)	-0.00012 (1)	-0.00002 (0)	-0.00002 (0)	-0.00004 (1)	-0.00004 (1)
	2	-0.00002 (1)	-0.00056 (1)	-0.00001 (1)	-0.00002 (1)	-0.00004 (0)	-0.00013 (1)	-0.00002 (0)	-0.00001 (1)	-0.00003 (1)	-0.00002 (1)
10%	3	-0.00002 (1)	-0.00064 (1)	-0.00000 (1)	-0.00004 (1)	-0.00002 (0)	-0.00011 (1)	-0.00000 (1)	-0.00001 (0)	-0.00001 (1)	-0.00001 (1)
	4	-0.00002 (1)	-0.00084 (1)	0.00002 (1)	-0.00004 (1)	-0.00001 (0)	-0.00010 (1)	0.00000 (0)	0.00001 (0)	-0.00001 (1)	-0.00001 (1)
	5	-0.00003 (1)	-0.00061 (1)	-0.00001 (0)	-0.00002 (1)	-0.00003 (0)	-0.00011 (1)	-0.00001 (1)	-0.00000 (1)	-0.00003 (1)	-0.00002 (1)
	1	-0.00005 (1)	-0.00241 (1)	-0.00004 (0)	-0.00016 (1)	-0.00003 (1)	-0.00008 (1)	-0.00003 (0)	-0.00003 (0)	-0.00001 (1)	-0.00003 (1)
	2	-0.00005 (1)	-0.00212 (1)	-0.00002 (0)	-0.00010 (1)	-0.00001 (0)	-0.00014 (1)	-0.00002 (0)	-0.00002 (0)	-0.00002 (0)	-0.00000 (1)
15%	3	-0.00004 (1)	-0.00190 (1)	-0.00002 (0)	-0.00012 (1)	-0.00002 (0)	-0.00011 (1)	-0.00005 (0)	-0.00003 (0)	-0.00002 (0)	0.00000 (0)
	4	-0.00003 (1)	-0.00295 (1)	-0.00000 (0)	-0.00005 (1)	-0.00004 (0)	-0.00014 (1)	-0.00001 (1)	-0.00001 (1)	-0.00004 (0)	-0.00001 (1)
	5	-0.00004 (1)	-0.00854 (1)	-0.00003 (0)	-0.00007 (1)	-0.00003 (1)	-0.00014 (1)	-0.00003 (0)	-0.00002 (1)	-0.00004 (0)	-0.00002 (0)
	1	-0.00001 (1)	-0.00229 (1)	-0.00001 (1)	-0.00003 (1)	-0.00002 (0)	-0.00009 (1)	-0.00001 (0)	-0.00002 (1)	-0.00005 (0)	-0.00001 (1)
	2	-0.00002 (1)	-0.00267 (1)	-0.00001 (1)	-0.00006 (1)	-0.00001 (0)	-0.00008 (1)	-0.00002 (1)	-0.00001 (0)	-0.00004 (0)	-0.00001 (1)
20%	3	-0.00000 (0)	-0.00312 (1)	-0.00000 (0)	-0.00005 (1)	-0.00003 (0)	-0.00009 (1)	-0.00002 (0)	-0.00001 (1)	-0.00005 (0)	-0.00002 (1)
	4	-0.00001 (0)	-0.00247 (1)	-0.00002 (1)	-0.00004 (1)	-0.00002 (0)	-0.00011 (1)	-0.00000 (1)	-0.00001 (1)	-0.00004 (0)	-0.00004 (1)
	5	-0.00003 (1)	-0.00202 (1)	-0.00000 (0)	-0.00007 (1)	-0.00001 (0)	-0.00007 (1)	-0.00000 (0)	-0.00002 (0)	-0.00005 (0)	-0.00001 (0)

pm10

RTT	Best Comb	Hour	PM10_in	RH_in	T_in	P_in	PM10_out	RH_out	T_out	num_year	num_week
	1	-0.00005 (1)	-0.00136 (1)	-0.00003 (1)	-0.00005 (1)	-0.00001 (0)	-0.00009 (1)	-0.00002 (0)	-0.00000 (0)	-0.00003 (0)	-0.00002 (1)
5%	2	-0.00002 (1)	-0.00114 (1)	-0.00002 (1)	-0.00003 (1)	-0.00003 (0)	-0.00009 (1)	-0.00000 (0)	0.00000 (1)	-0.00003 (0)	-0.00001 (1)

	3	-0.00003 (1)	-0.00103 (1)	-0.00000 (1)	-0.00006 (1)	-0.00002 (0)	-0.00010 (1)	0.00000 (1)	-0.00002 (1)	-0.00003 (0)	-0.00004 (1)
	4	-0.00002 (1)	-0.00108 (1)	0.00000 (0)	-0.00004 (1)	-0.00003 (0)	-0.00009 (1)	-0.00002 (1)	-0.00001 (1)	-0.00002 (0)	-0.00001 (1)
	5	-0.00004 (1)	-0.00148 (1)	-0.00003 (1)	-0.00004 (1)	-0.00004 (0)	-0.00011 (1)	-0.00004 (0)	-0.00001 (1)	-0.00003 (0)	0.00001 (0)
	1	-0.00005 (1)	-0.00097 (1)	-0.00002 (1)	-0.00006 (1)	-0.00002 (1)	-0.00013 (1)	-0.00003 (0)	-0.00003 (0)	-0.00002 (1)	-0.00006 (1)
	2	-0.00001 (1)	-0.00084 (1)	-0.00001 (0)	-0.00004 (1)	-0.00002 (0)	-0.00013 (1)	-0.00002 (0)	-0.00000 (1)	-0.00002 (1)	-0.00003 (1)
10%	3	-0.00003 (1)	-0.00093 (1)	-0.00001 (0)	-0.00004 (1)	-0.00002 (0)	-0.00010 (1)	-0.00000 (0)	0.00000 (0)	-0.00002 (1)	-0.00001 (1)
	4	-0.00003 (1)	-0.00086 (1)	-0.00000 (0)	-0.00005 (1)	-0.00004 (0)	-0.00010 (1)	0.00000 (1)	-0.00002 (0)	-0.00003 (1)	-0.00002 (1)
	5	-0.00003 (1)	-0.00081 (1)	0.00000 (1)	-0.00005 (1)	-0.00003 (0)	-0.00011 (1)	-0.00001 (1)	-0.00000 (0)	-0.00001 (1)	-0.00001 (1)
	1	-0.00004 (1)	-0.00168 (1)	-0.00002 (0)	-0.00010 (1)	-0.00007 (0)	-0.00013 (1)	-0.00005 (0)	-0.00001 (1)	-0.00003 (1)	-0.00005 (1)
	2	-0.00003 (1)	-0.00301 (1)	-0.00007 (0)	-0.00011 (1)	-0.00003 (1)	-0.00014 (1)	-0.00006 (0)	-0.00002 (1)	-0.00007 (1)	-0.00006 (0)
15%	3	-0.00004 (1)	-0.00225 (1)	-0.00004 (0)	-0.00018 (1)	-0.00002 (0)	-0.00011 (1)	-0.00007 (0)	0.00001 (0)	-0.00004 (1)	-0.00002 (1)
	4	-0.00005 (1)	-0.00680 (1)	-0.00002 (0)	-0.00013 (1)	-0.00003 (1)	-0.00016 (1)	-0.00003 (0)	-0.00002 (0)	-0.00002 (0)	-0.00003 (1)
	5	-0.00003 (1)	-0.00295 (1)	-0.00001 (1)	-0.00007 (1)	-0.00003 (0)	-0.00012 (1)	-0.00003 (0)	-0.00001 (1)	-0.00000 (0)	-0.00002 (1)
	1	-0.00005 (1)	-0.00359 (1)	-0.00001 (1)	-0.00009 (1)	-0.00000 (1)	-0.00012 (1)	-0.00004 (0)	-0.00002 (0)	-0.00005 (0)	-0.00003 (1)
	2	-0.00005 (1)	-0.00203 (1)	-0.00001 (1)	-0.00009 (1)	0.00000 (0)	-0.00010 (1)	-0.00003 (0)	-0.00002 (0)	-0.00004 (0)	-0.00003 (1)
20%	3	-0.00004 (1)	-0.00276 (1)	0.00001 (0)	-0.00010 (1)	-0.00001 (0)	-0.00009 (1)	-0.00001 (0)	-0.00002 (0)	-0.00002 (0)	-0.00003 (1)
	4	-0.00003 (1)	-0.00434 (1)	0.00001 (0)	-0.00010 (1)	0.00001 (1)	-0.00011 (1)	-0.00001 (0)	-0.00003 (0)	-0.00007 (0)	-0.00002 (1)
	5	-0.00002 (1)	-0.00319 (1)	-0.00002 (0)	-0.00006 (1)	-0.00002 (0)	-0.00013 (1)	-0.00002 (0)	-0.00002 (1)	-0.00006 (0)	-0.00001 (0)

Table 9

References:

1. [6] Feng Z, Yu CW, Cao SJ. Fast prediction for indoor environment: models assessment. (2019): 727-730.
2. [6.1] Wei, W., Ramalho, O., Malingre, L., Sivanantham, S., Little, J. C., & Mandin, C. (2019). Machine learning and statistical models for predicting indoor air quality. Indoor Air, 29(5), 704-726.
3. [7] Sharma PK, Mondal A, Jaiswal S, Saha M, Nandi S, De T, Saha S. IndoAirSense: A framework for indoor air quality estimation and forecasting. Atmospheric Pollution Research. 2021 Jan 1;12(1):10-22.
4. [8] Lagesse B, Wang S, Larson TV, Kim AA. Predicting PM2. 5 in Well-Mixed Indoor Air for a Large Office Building Using Regression and Artificial Neural Network Models. Environmental Science & Technology. 2020 Nov 17;54(23):15320-8.
5. [8.1] Liu, DR., Hsu, YK., Chen, HY. et al. Air pollution prediction based on factory-aware attentional LSTM neural network. Computing 103, 75–98 (2021). https://doi.org/10.1007/s00607-020-00849-y
6. [9] Li, Z., Tong, X., Ho, J. M. W., Kwok, T. C., Dong, G., Ho, K. F., & Yim, S. H. L. (2021). A practical framework for predicting residential indoor PM2. 5 concentration using land-use regression and machine learning methods. Chemosphere, 265, 129140.
7. Wang, Z., Hong, T., & Piette, M. A. (2020). Building thermal load prediction through shallow machine learning and deep learning. Applied Energy, 263, 114683.
8. [0] Zivot E., Wang J. (2003) Rolling Analysis of Time Series. In: Modeling Financial Time Series with S-Plus®. Springer, New York, NY. https://doi.org/10.1007/978-0-387-21763-5_9

9. [1] Marill, K. A. (2004). Advanced statistics: linear regression, part I: simple linear regression. *Academic emergency medicine*, *11*(1), 87-93.
10. [2] Suthaharan, S. (2016). Support vector machine. In *Machine learning models and algorithms for big data classification* (pp. 207-235). Springer, Boston, MA.
11. [3]Segal MR. Machine learning benchmarks and random forest regression.
12. [4] Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, *7*, 21.
13. [4.1] Dey R, Salem FM. Gate-variants of gated recurrent unit (GRU) neural networks. In2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS) 2017 Aug 6 (pp. 1597-1600). IEEE.
14. [5] Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation. 1997 Nov 15;9(8):1735-80.
15. [5.1] Huovila, P., Ala-Juusela, M., Melchert, L., Pouffary, S., Cheng, C. C., Ürge-Vorsatz, D., ... & Graham, P. (2009). Buildings and climate change: Summary for Decision-Makers.
16. [5.2]Zhang, Z., Zeng, Y., & Yan, K. (2021). A hybrid deep learning technology for PM 2.5 air quality forecasting. Environmental Science and Pollution Research, 1-14.
17. [9]Li, Z., Tong, X., Ho, J. M. W., Kwok, T. C., Dong, G., Ho, K. F., & Yim, S. H. L. (2021). A practical framework for predicting residential indoor PM2. 5 concentration using land-use regression and machine learning methods. Chemosphere, 265, 129140.
18. [10.1] Abdi H, Williams LJ. Principal component analysis. Wiley interdisciplinary reviews: computational statistics. 2010 Jul;2(4):433-59.
19. [10.2] Cieslak MC, Castelfranco AM, Roncalli V, Lenz PH, Hartline DK. t-Distributed Stochastic Neighbor Embedding (t-SNE): A tool for eco-physiological transcriptomic analysis. Marine genomics. 2020 Jun 1;51:100723.
20. [16] https://files.airnowtech.org/
21. [17] https://www.purpleair.com/sensorlist?key=71XU48F19Q4YGD4F&show=22463
22. [18] http://beacon.berkeley.edu/
23. [18.4] Browne, M. (2003). Cross-Validation Methods. https://doi.org/10.1006/jmps.1999.1279