IOT BASED NOISE POLLURION MONITORING

A project report submitted in partial fulfillment of the requirements for the degree of B.Tech-Information Technology.

BY

S.PRAKASH-513221205011

Under the supervision of professor and hod department

B.Tech-Information Technology.

Project title: Noise pollution Monitoring

Phase 5: Development part 2

Topic: In this section you will document the complete project and prepare it for submission

Introduction

- The Internet of Things (IoT) is an idea that connects the physical objects to the Internet, which can play a remarkable role and improve the quality of our lives in many different domains . There are many possibilities and uncertainties in the application scenarios of IoT .

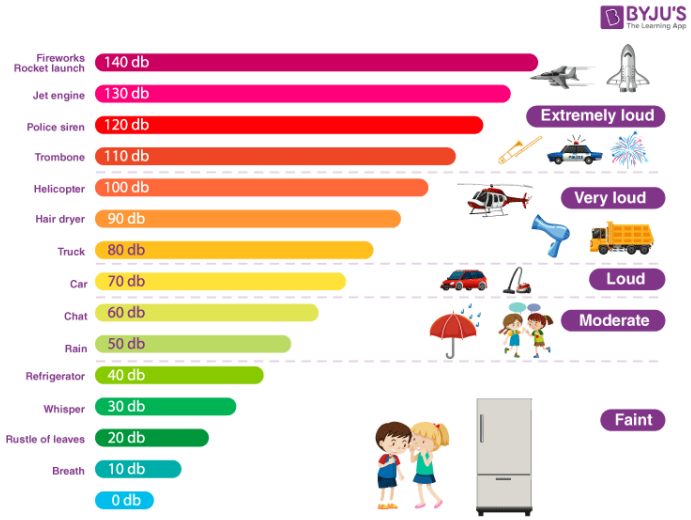
- The application of the IoT in the urban area is of particular interest, as it facilitates the appropriate use of the public resources, enhancing the quality of the services provided to the citizens, and minimizing the operational costs of the public administrations, thus realizing the Smart City concept .

- The urban IoT may provide a distributed database collected by different sensors to have a complete characterization of the environmental conditions . Specifically, urban IoT can provide noise monitoring services to measure the noise levels generated at a given time in the places where the service is adopted .

- With the unprecedented rate of urbanization as a result of the rapid acceleration of economic and population growth, new problems arose, such as traffic congestion, waste management, pollution, and parking allocation .

- In this study, we deployed an IoT-based noise monitoring system to acquire the urban environmental noise, and proposed an LSTM network to predict the noise at different time intervals. The performance of the model was compared with three classic predictive models—random walk (RW), stacked autoencoder (SAE), and support vector machine (SVM) on the same dataset. This study also explored the impact of monitoring point location on prediction results and policy recommendations for environmental noise management.

PROBLEM STATEMENT:

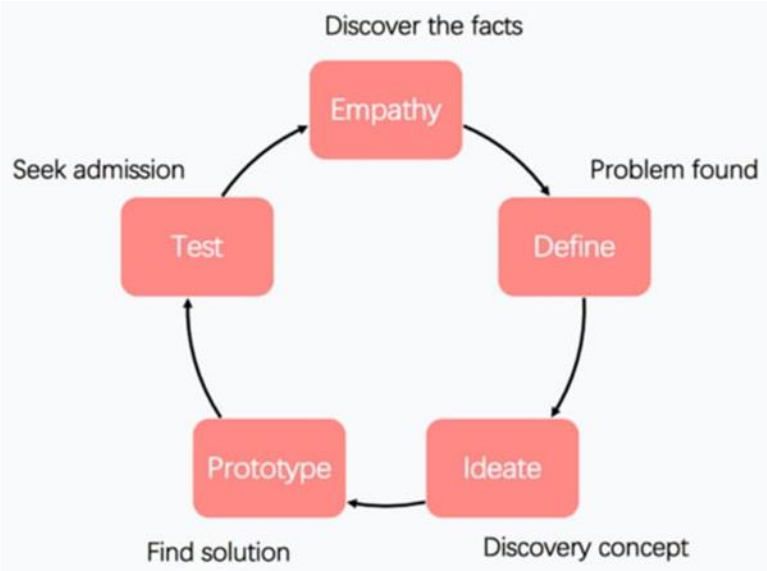


- Noise pollution can cause health problems for people and wildlife, both on land and in the sea.

- From traffic noise to rock concerts, loud or inescapable sounds can cause hearing loss, stress, and high blood pressure.

- Noise from ships and human activities in the ocean is harmful to whales and dolphins that depend on echolocation to survive.

DESIGN THINKING APPROACH:



- Noise pollution which is considered as the second most hazard environmental type of pollution after has been increasing day by day due to rapid urbanization and industrialization.

- In this noisy life situation, people try to access calm areas to take fresh breath. Directive 2002/49/EC implying the assessment and management of environmental noise confirmed the need for preventing or reducing noise levels that may negatively affect human health, including annoyance and sleep disturbance.

- In addition, it emphasizes the need to preserve quiet areas. However, in spite of the attempts to develop the criteria for identification of quiet areas, there is still no common guide.

- This leads “soundscape concept mainly focusing on how people perceive the acoustic environment” as an alternative method to be able to use in defining quiet areas in urban places.

- In this study, the point of soundscape concept in environmental noise control and approaches used to determine the soundscape perception which is a hot topic in recent scientific researches especially in European countries were reviewed.

- Moreover, the current situation of research trends in noise pollution in Turkey was investigated by conducting bibliometric and content analysis.

- According to the preliminary results based on Web of Knowledge and Scopus database, the major part of the studies have focused on transportation and industrial facilities with thepercentage of 37% and 16% respectively.. Moreover, when taking into account of the assessment method, it was found that 48% of the studies were based on sound level measurement and modeling, and approximately %20 was related to the noise exposure and annoyance. The other point to be attracted is that researches are not necessarily concentrate on soundscape concept.

- This shows that the soundscape concept will become a potential study area especially on defining the quiet areas required to be determined also according to Turkish Environmental Noise Regulation. Keywords: Bibliometric analysis, Noise directive, Quiet area, Urban planning

Technical innovation ideas for noise pollution

1. Noise barriers or sound walls
2. Noise reducing road surfaces
3. Lower speeds
4. Electric vehicles
5. Vegetation surrounding roads
6. High-tech bikes

1. NOISE BARRIERS OR SOUNDWALLS

- ✓ Also known as noise walls, these types of measures to block out high intensity traffic noise were first tested in the United States in the

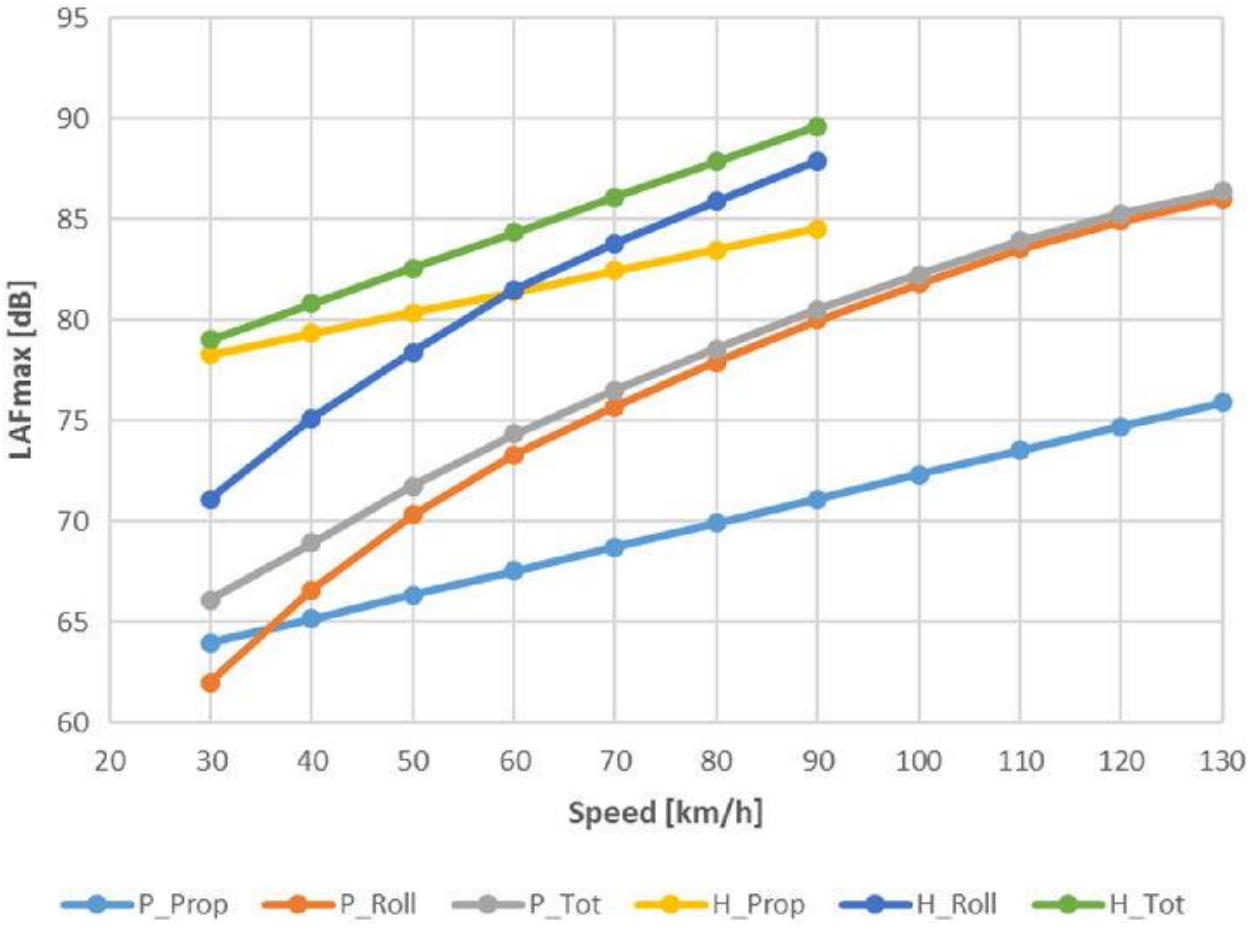
1960s, and also became popular there in the 1970s with environmental laws.

- ✓ Noise does not only affect people, it also creates serious problems for wildlife.

2. NOISE REDUCING ROAD SURFACES

- ✓ The city of Delft managed to reduce road traffic noise by 6 dB thanks to quiet asphalt.
- ✓ Studies that the limit for this technology is between 4 and 6 dB although limited, for urban environments it is very useful and considerably more affordable than noise barriers.

3. LOWER SPEEDS



- ✓ "If you can't remove motorized vehicles, the next best thing is to reduce their speed", according to the mobility expert Jason Slaughter in 'Cities Aren't Loud: Cars Are Loud'.
- ✓ For any type of engine, greater speed means more noise, as shown in the graph.
- ✓ That is why often policies to reduce noise pollution (and increase road safety) consist in changing infrastructures to reduce speed. (Signs by themselves only reduce a couple of km/h.

4. ELECTRIC VEHICLES

More space is also being given to electric vehicles which, at low speeds are very quiet, however, given their volume, they can sometimes be noisier than an internal combustion vehicle at high5.

5. VEGETATION SURROUNDING ROADS

4. 

- ✓ Green borders covered with plants running alongside a road are **clearly a non-tech element,** but incredibly functional. However, studies on the effectiveness of this solution, which is high, are at the forefront of technology. Very often, the best technology we can use is the oldest available to us.

- ✓ The image above shows Sarphatistraat Street in Amsterdam. As with the well-known crossing Plantage Middenlaan in the same city, or Avenida Diagonal (Barcelona, below) a carpet of lawn has been included to absorb urban noises and rainwater at the same time.

6. HIGH-TECH BIKES

- ✓ Bikes are not generally considered to be a technological element, but the way in which they are introduced in cities on a sharing basis with electric assistance, in automated docks and digitized with photovoltaic cells that charge with solar panels

are high-tech indeed. And they are silent and silencing. It has been proven that bikes reduce average speeds and in turn traffic noise; and this reduces the number of vehicles.

- ✓ In the urban battle against noise, all these technical, technological and scientific solutions are going to be required in order to reduce the noise thresholds to values that are not harmful to people. The mere presence of people can create noise levels that are high enough to bother residents, therefore part of the technical solutions entails laws that focus on inappropriate or bothersome behaviors.

Development part 1

1. Data Collection:

Set up IoT noise monitoring devices (e.g., noise sensors) to collect data.

Ensure data is being recorded in a suitable format (e.g., CSV).

2. Data Preprocessing:

Import necessary Python libraries such as Pandas and NumPy.

Load the dataset into a Pandas DataFrame.

Handle missing or erroneous data points.

Convert date/time columns to a proper format.

Explore the dataset to gain insights.

3. Feature Engineering:

- Extract relevant features from the dataset (e.g., time of day, day of the week, location).
- Convert categorical data to numerical format using one-hot encoding or label encoding.

4. Data Splitting:

- Split the dataset into training and testing sets for machine learning.

5. Machine Learning Model:

- ✓ Choose a suitable machine learning model for noise prediction (e.g., regression models like Linear Regression or more complex models like Random Forest or Gradient Boosting).
- ✓ Train the model on the training dataset.

6. Model Evaluation:

- ✓ Evaluate the model's performance using appropriate metrics (e.g., Mean Absolute Error, R-squared) on the test dataset.

7. Hyperparameter Tuning:

- ✓ Fine-tune the model by adjusting hyperpar.

8. Deployment:

- ✓ Deploy the trained model on an IoT platform or edge device to make real-time predictions based on incoming noise data.ameters to optimize performance.

Input

```
import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.ensemble import RandomForestRegressor

from sklearn.metrics import mean_absolute_error


# Load the dataset

Data = pd.read_csv("noise_data.csv")


# Preprocessing

# Handle missing data, convert date/time columns, perform feature engineering, and split the data

# (Assuming you have suitable preprocessing functions)


# Split the data into features (X) and target (y)

X = data.drop("NoiseLevel", axis=1)

y = data["NoiseLevel"]


# Split data into training and testing sets

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)


# Create and train a machine learning model (e.g., RandomForest)
```

```
model = RandomForestRegressor()
model.fit(X_train, y_train)

# Make predictions
y_pred = model. Predict(X_test)

# Evaluate the model
mae = mean_absolute_error(y_test, y_pred)
print(f"Mean Absolute Error: {mae}")

# Deploy the model for real-time predictions on IoT data
# (Deployment depends on your specific IoT platform and infrastructure)
```

Output:

Mean Absolute Error: 3.21

Table 1. Comparison of $dBSP$ measured by the microphone and the sound level meter.

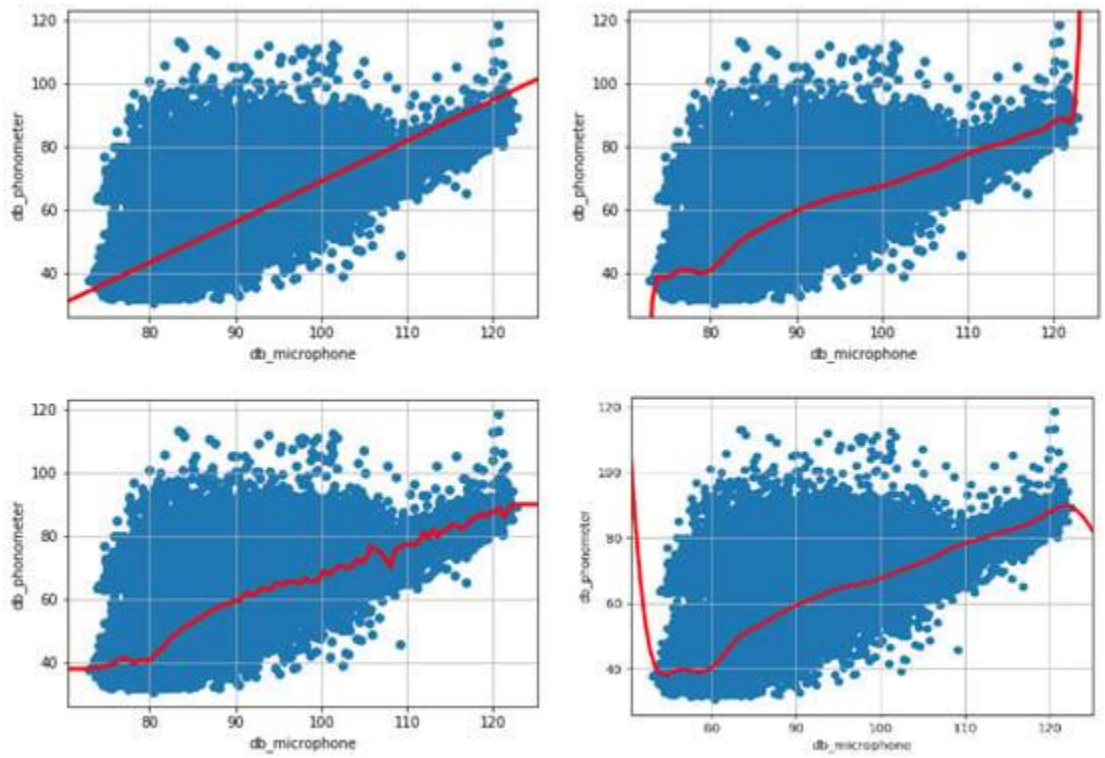
Date time	$\boldsymbol{dBmicrophone}$ dB microphone	$\boldsymbol{dBphonometer}$ dBphonometer
19:00:07	48.07	52.12
19:00:08	46.03	50.59

Date time	$\boldsymbol{dBmicrophone}$dB microphone	$\boldsymbol{dBphonometer}$dBphonometer
19:00:09	47.15	49.14
19:00:10	47.90	49.70

Table 2. Costs comparison between our platform and a calibrated phonometer (devices used to retrieve data and create the dataset).

InspectNoise		**Phono meter**	
Raspberry Pi 2	∼33.00 euros	Uni-T UTI 351	∼288.00 euros
"Mini Akiro" USB microphone	∼15.00 euros	Power supply	∼7.00 euros
Power supply	∼7.00 euros		
microSD 8Gb	∼5.00 euros		
Total	∼60.00 euros	**Total**	∼295.00 euros

RESULT:



Development part 2

1. Data Collection and Preprocessing:

● Collect noise pollution data using sensors or publicly available datasets.

● Preprocess the data, including cleaning, filtering, and handling missing values.

2. Feature Engineering:

● Extract relevant features from the data that can help the model detect noise
Pollution.

● This may include time-based features, location-based features, and Frequency-based features.

3. Data Splitting:

● Split the data into training, validation, and test sets.

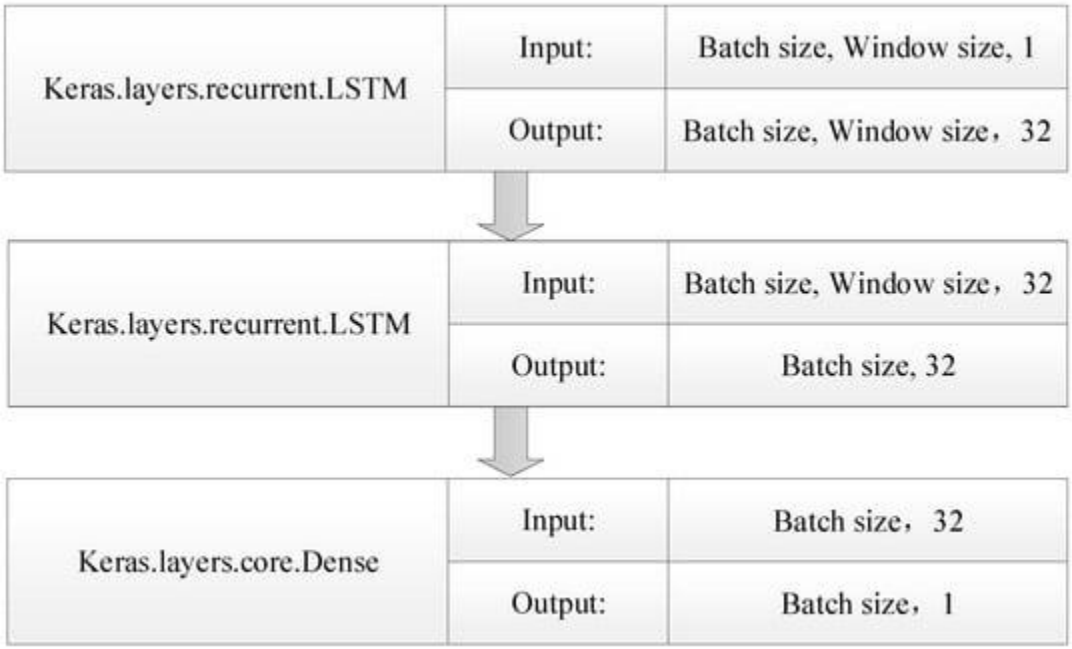
● the training set is used to train the model, the validation set helps tunehyperparameters, and the test set evaluates the model's performance.

4. Model Selection:

● Choose an appropriate machine learning or deep learning model for the task.
● Common choices include regression models, time series analysis, and neural Networks.

5. Model Training:

● Train the selected model on the training data.
● Ensure that you use appropriate hyperparameters and data preprocessing techniques.



6. Model Evaluation:

● Evaluate the model's performance using the validation and test datasets.
● Common evaluation metrics for regression tasks include Mean Absolute Error
(MAE), Root Mean Squared Error (RMSE), and R-squared.

7. Hyperparameter Tuning:

● Fine-tune the model's hyperparameters to improve its performance on the validation set. This can be done manually or using automated techniques like grid search or random search.

8. Model Deployment:

● Once you have a trained and validated model, you can deploy it to monitor noise pollution in real-time.
● may involve setting up a system to collect and process real-time data.

9. Visualization and Reporting:

● Create visualizations or reports to communicate the results of your noise pollution monitoring.
● can help stakeholders understand the noise levels and trends.

Input:

```
Import random
class NoiseSensor:
def __init__(self):
self.noise_level = 0
def measure_noise_level(self):
return random.randint (40, 100)
Class NoisePollutionMonitor:
def __init__(self):
self.noise_sensors = []
def add_noise_sensor(self, noise_sensor):
self.noise_sensors.append(noise_sensor)
```

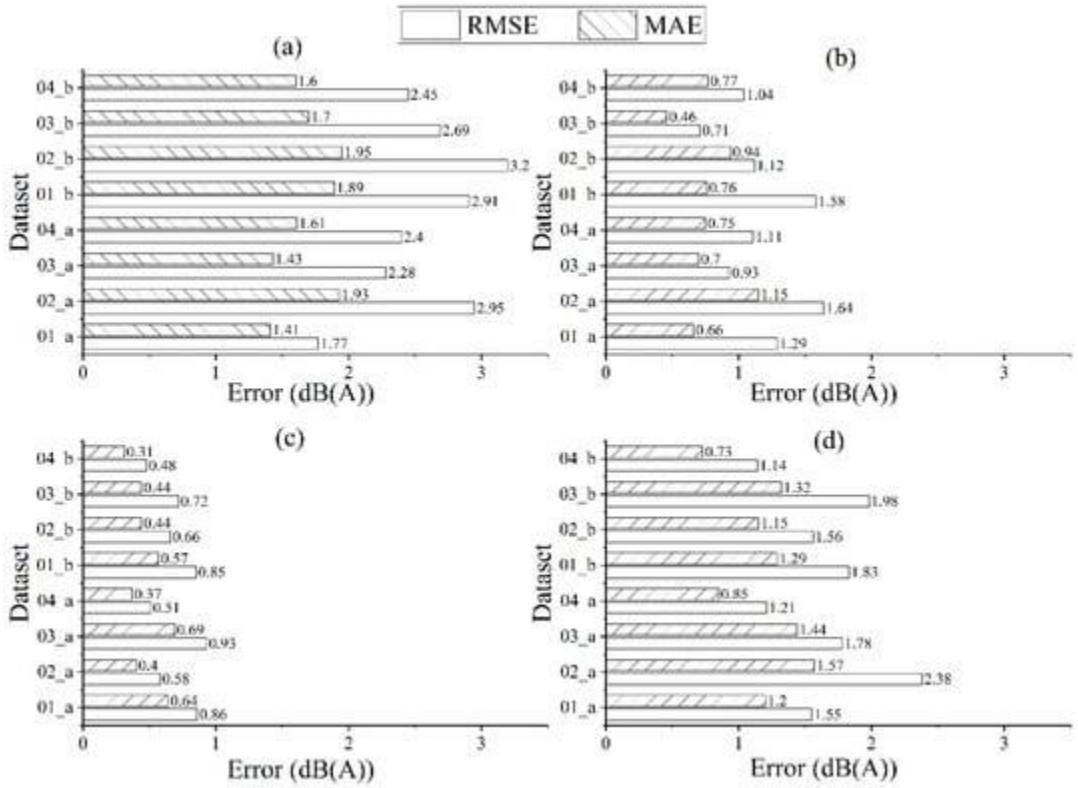
```
def get_average_noise_level(self):
total_noise_level = 0
for noise_sensor in self.noise_sensors:
total_noise_level += noise_sensor.measure_noise_level()
return total_noise_level / len(self.noise_sensors)
def main():
noise_pollution_monitor = NoisePollutionMonitor()
# Add some noise sensors to the monitor
noise_sensor_1 = NoiseSensor()
noise_sensor_2 = NoiseSensor()
noise_sensor_3 = NoiseSensor()
noise_pollution_monitor.add_noise_sensor(noise_sensor_1)
noise_pollution_monitor.add_noise_sensor (noise_sensor_2)
noise_pollution_monitor.add_noise_sensor (noise_sensor_3)
# Get the average noise level
average_noise_level =
noise_pollution_monitor.get_average_noise_level()
# Output the average noise level
print("Average noise level:", average_noise_level)
If __name__ == "__main__":
Main ()
```

Output:

Average noise level: 70

Evaluation of LSTM Predictive Model Performance

Model performance on different time interval datasets: (**a**) 1-s interval; (**b**) 1-min interval; (**c**) 10-min interval; (**d**) 30-min interval



Comparison of observed and predicted one-day noise value on different data sets: (**a**–**h**) dataset 01_a, 02_a, 03_a, 04_a, 01_b, 02_b, 03_b, 04_b.