**College of Engineering (COE)**

**National Chung Cheng University**

**Report of TEEP@ASIAPLUS2023 Intern Research Program**

| | | | |
|---|---|---|---|
| **Student** | Nishad Dhuri | **Faculty Member** | Dr. Pao-Ann Hsiung |
| **Department** | | Department of Computer Science and Information Engineering | |
| **Research Period**<br>2023/07/02 to 2023/12/28 | | **Date of Report Submission**<br>2024/07/18 | |
| **Report Title** | Multimodal Federated Learning for Smart Cities | | |
| **Highlights of Report** | This study developed an innovative method for estimating the PM concentrations in the air. It started with a collection of the atmospheric data for multiple stations throughout Taiwan and went on to calculate the PM values using a model developed while following the fed-multimodal framework. The crucial part was creating prediction models and refining them with past data to precisely calculate emissions from numerical and image data. These models, which used machine learning techniques, predict the PM concentrations in the atmosphere using multimodal image and numerical data. | | |
| **Signature of Report** | | **Date** | |
| **Mentor Review Comment** | □ Excellent □Good □Average □Poor | | |
| **Signature of Mentor** | | **Date** | |

College of Engineering

National Chung Cheng University

# Sustainability Assessment (SDG11) for Smart Cities

# Multimodal Federated Learning for Smart Cities

# Estimating PM concentration in air via multimodal data

*Nishad Dhuri*

*Department of Computer Science and Information Engineering*

*National Chung Cheng University, Chia-Yi 621, Taiwan*

## Abstract

This paper presents a novel model architecture for estimating PM2.5 and PM10 values using data from two modalities - numerical and images. Our model follows the fed-multimodal framework, and leverages GRUs, CNNs, residual connections and self attention mechanisms. The dataset consists of 15 meteorological features, and images for each hour of each day of the year. The model was trained on data from several stations across Taiwan, and yielded acceptable results on unseen data, showing promising generalization capabilities. We address the model overfitting problem via residual connections, and explored various ways to handle the data heterogeneity issues for our problem.

## Table of Contents

# 1. INTRODUCTION

The relationship between air quality and human quality of life and well-being is well-established. The deteriorating air quality is the cause of human illnesses and ecological destruction . The WHO report says that more than 8 million deaths are estimated to occur annually due to PM pollution in the world . Specifically, PM2.5 prediction and classification are essential to society because PM2.5 is a tiny particle with a diameter of less than 2.5 microns and transmits hazardous chemicals directly into the bloodstream and lungs, causing many diseases such as asthma, bronchitis, and other lung diseases . It has become a big reason why the government is always looking for good ways to measure and deal with the problem. The air quality index (AQI) is a standard unit utilized by various stations to measure and report air quality.

Recently, image-based air quality estimation has become a popular topic because researchers have realized that smartphone cameras make it feasible to acquire images. This will allow efficient monitoring of the PM concentrations, leading to better general health. This paper contributes towards this end by proposing a model that takes as input some general atmospheric measurements, and a picture. It also explores the various challenges and data heterogeneity problems encountered in the multimodal setup, and explores ways to address them.

# 2. Literature Review / Related Work

Fed-Multimodal : https://dl.acm.org/doi/pdf/10.1145/3580305.3599825
MM Learning : https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10041115
Residual Networks : https://arxiv.org/pdf/1512.03385
Unsupervised Pretraining : https://www.nature.com/articles/s41598-019-55320-6.pdf
MKL: https://jmlr.csail.mit.edu/papers/volume12/gonen11a/gonen11a.pdf
Transformers:
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10153685&tag=1
PM Estimation : https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9279204
Spatial and Context Attention Block :
https://www.sciencedirect.com/science/article/pii/S0048969720316910?via%3Dihub
Federated Learning : https://arxiv.org/pdf/2206.12949

# 3. Methodology

The research methodology involves a systematic process to estimate the PM concentrations at various meteorological stations. The data was sourced from Taiwan governments website, and consists of hourly images and numerical readings from various sensors for multiple stations. Following data collection, the images were preprocessed for efficient feature extraction from pretrained CNN based models. Exploratory Data Analysis (EDA) techniques provide an insightful overview of trends and patterns within the datasets. Deep learning techniques are employed to predict the PM values, enabling easy estimation of air quality. The methodology concludes with a rigorous evaluation of the models' performance, culminating in acceptable generalization and performance on unseen data.

## 3.1. Data Gathering

The data was gathered from Taiwan governments official website for air quality monitoring (https://airtw.moenv.gov.tw/ ). Data was obtained for the year of 2022, from February to December(the images are missing for January).

The numerical data is in the format of a csv file, consisting of 18 numerical features for each hour of each day. Some of the entries in the csv file are missing. The 18 numerical features provided are : AMB_TEMP, CH4, CO, NMHC, NO, NO2, NOx, O3,PM10,PM2.5,RAINFALL, RH,SO2,THC,WD_HR,WIND_DIR,WIND_SPEED,WS_HR. For many of the stations, the CH4, NMHC and THC values are missing.

The image data consists of missing and corrupted images. The images for each station are static and captured from the same location. The night time images are mostly in black and white for each station, with a few exceptions. The images for some stations are of bad quality, which resulted in poor performance on them (shown later).

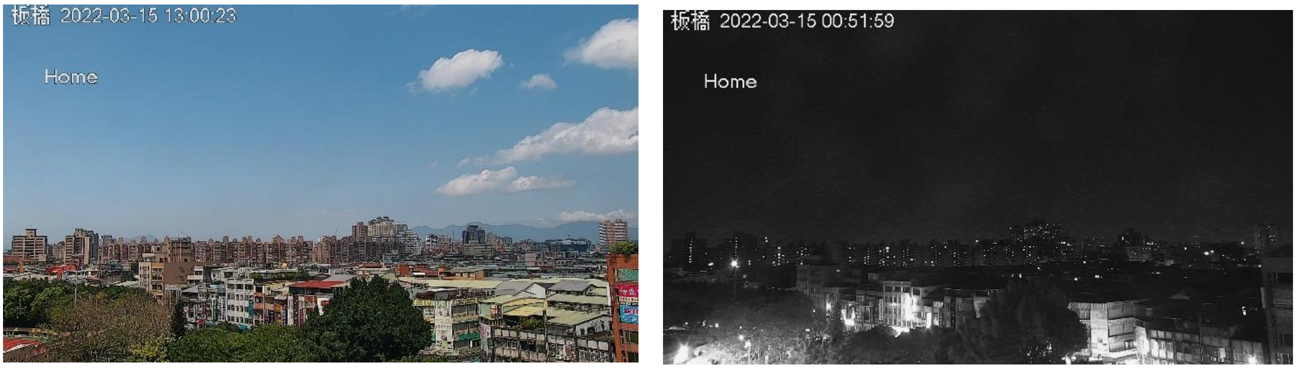
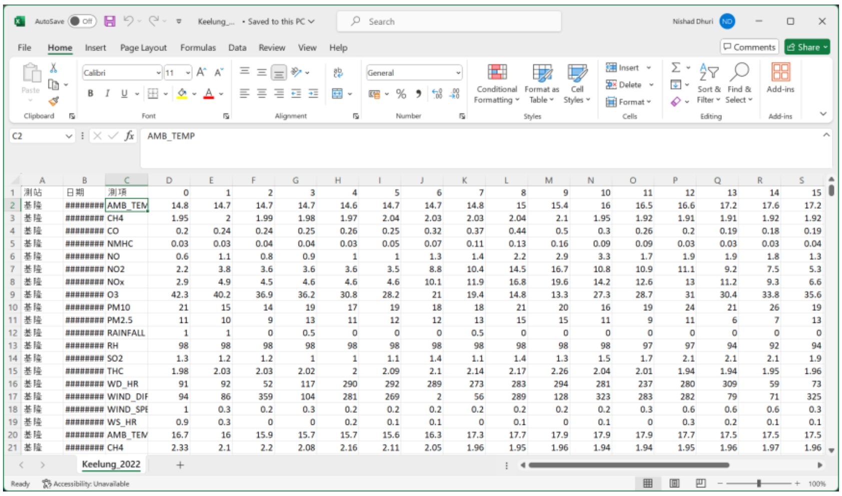


*Figure 1: Day and Night images for Banqiao*



| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 測站 | 日期 | 測項 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 2 | 基隆 | ######## | AMB_TEM | 14.8 | 14.7 | 14.7 | 14.7 | 14.6 | 14.7 | 14.7 | 14.8 | 15 | 15.4 | 16 | 16.5 | 16.6 | 17.2 | 17.6 | 17.2 |
| 3 | 基隆 | ######## | CH4 | 1.95 | 2 | 1.99 | 1.98 | 1.97 | 2.04 | 2.03 | 2.03 | 2.04 | 2.1 | 1.95 | 1.92 | 1.91 | 1.91 | 1.92 | 1.92 |
| 4 | 基隆 | ######## | CO | 0.2 | 0.24 | 0.24 | 0.25 | 0.26 | 0.25 | 0.32 | 0.37 | 0.44 | 0.5 | 0.3 | 0.26 | 0.2 | 0.19 | 0.18 | 0.19 |
| 5 | 基隆 | ######## | NMHC | 0.03 | 0.03 | 0.04 | 0.04 | 0.03 | 0.05 | 0.07 | 0.11 | 0.13 | 0.16 | 0.09 | 0.09 | 0.03 | 0.03 | 0.03 | 0.04 |
| 6 | 基隆 | ######## | NO | 0.6 | 1.1 | 0.8 | 0.9 | 1 | 1 | 1.3 | 1.4 | 2.2 | 2.9 | 3.3 | 1.7 | 1.9 | 1.9 | 1.8 | 1.3 |
| 7 | 基隆 | ######## | NO2 | 2.2 | 3.8 | 3.6 | 3.6 | 3.6 | 3.5 | 8.8 | 10.4 | 14.5 | 16.7 | 10.8 | 10.9 | 11.1 | 9.2 | 7.5 | 5.3 |
| 8 | 基隆 | ######## | NOx | 2.9 | 4.9 | 4.5 | 4.6 | 4.6 | 4.6 | 10.1 | 11.9 | 16.8 | 19.6 | 14.2 | 12.6 | 13 | 11.2 | 9.3 | 6.6 |
| 9 | 基隆 | ######## | O3 | 42.3 | 40.2 | 36.9 | 36.2 | 30.8 | 28.2 | 21 | 19.4 | 14.8 | 13.3 | 27.3 | 28.7 | 31 | 30.4 | 33.8 | 35.6 |
| 10 | 基隆 | ######## | PM10 | 21 | 15 | 14 | 19 | 17 | 19 | 18 | 18 | 21 | 20 | 16 | 19 | 24 | 21 | 26 | 19 |
| 11 | 基隆 | ######## | PM2.5 | 11 | 10 | 9 | 13 | 11 | 12 | 12 | 13 | 15 | 15 | 11 | 9 | 11 | 6 | 7 | 13 |
| 12 | 基隆 | ######## | RAINFALL | 1 | 1 | 0 | 0.5 | 0 | 0 | 0 | 0.5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | 基隆 | ######## | RH | 98 | 98 | 98 | 98 | 98 | 98 | 98 | 98 | 98 | 98 | 98 | 97 | 97 | 94 | 92 | 94 |
| 14 | 基隆 | ######## | SO2 | 1.3 | 1.2 | 1.2 | 1 | 1 | 1.1 | 1.4 | 1.1 | 1.4 | 1.3 | 1.5 | 1.7 | 2.1 | 2.1 | 2.1 | 1.9 |
| 15 | 基隆 | ######## | THC | 1.98 | 2.03 | 2.03 | 2.02 | 2 | 2.09 | 2.1 | 2.14 | 2.17 | 2.26 | 2.04 | 2.01 | 1.94 | 1.94 | 1.95 | 1.96 |
| 16 | 基隆 | ######## | WD_HR | 91 | 92 | 52 | 117 | 290 | 292 | 289 | 273 | 283 | 294 | 281 | 237 | 280 | 309 | 59 | 73 |
| 17 | 基隆 | ######## | WIND_DIF | 94 | 86 | 359 | 104 | 281 | 269 | 2 | 56 | 289 | 128 | 323 | 283 | 282 | 79 | 71 | 325 |
| 18 | 基隆 | ######## | WIND_SPE | 1 | 0.3 | 0.2 | 0.3 | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 | 0.3 | 0.6 | 0.6 | 0.6 | 0.3 |
| 19 | 基隆 | ######## | WS_HR | 0.9 | 0.3 | 0 | 0 | 0.2 | 0.1 | 0.1 | 0 | 0.1 | 0 | 0.1 | 0 | 0.3 | 0.2 | 0.1 | 0.1 |
| 20 | 基隆 | ######## | AMB_TEM | 16.7 | 16 | 15.9 | 15.7 | 15.7 | 15.6 | 16.3 | 17.3 | 17.7 | 17.9 | 17.9 | 17.9 | 17.7 | 17.5 | 17.5 | 17.5 |
| 21 | 基隆 | ######## | CH4 | 2.33 | 2.1 | 2.2 | 2.08 | 2.16 | 2.11 | 2.05 | 1.96 | 1.95 | 1.96 | 1.94 | 1.94 | 1.95 | 1.96 | 1.97 | 1.96 |

*Figure 2: Numerical data for Keelung*

## 3.2. Data Processing and Observations

Upon collecting extensive data covering multiple stations, our initial step involved selecting and retaining pertinent information crucial for our research goals.

The numerical data was restructured in such a way that each row gives us the numerical features for an hour of the day. This makes it easier to read from the dataframe. The missing numerical values were filled by taking an average for that field over the entire year's data.

The images were preprocessed, and were converted to 224x224 dimensions, as this is the input dimension required for pretrained large image models like Resnet and MobileNet . The 224x224 images were generated in two ways: scaling the images, or cropping the central part. The cropped images generated better results, the cause for this could be that the MobileNet model was originally trained on cropped data. The missing or corrupted images were replaced with the images having the closest numerical data.

There were several observations made from the dataset :

- There is a lot of variation in pm values throughout the day. This could be the reason why the model successfully predicts the long term PM conc patterns, but fails to do so in a small interval.
- The dataset suffers from various different data heterogeneity issues. There is a large difference between the day time and night time images (colour and black and white). For each station, the images have static structures (buildings, trees etc) which might introduce some bias.
- Some numerical features like Relative Humidity and Rainfall don't have a strong co-relation with pm values, but the presence of moisture and rain adds a blur to the image, which can be mistaken as particulate matter, so these features need to be kept.

- The CH4, NHMC and THC values are missing for most of the stations, but they can be safely dropped, as CO has a strong co-relation with them, effectively representing them.

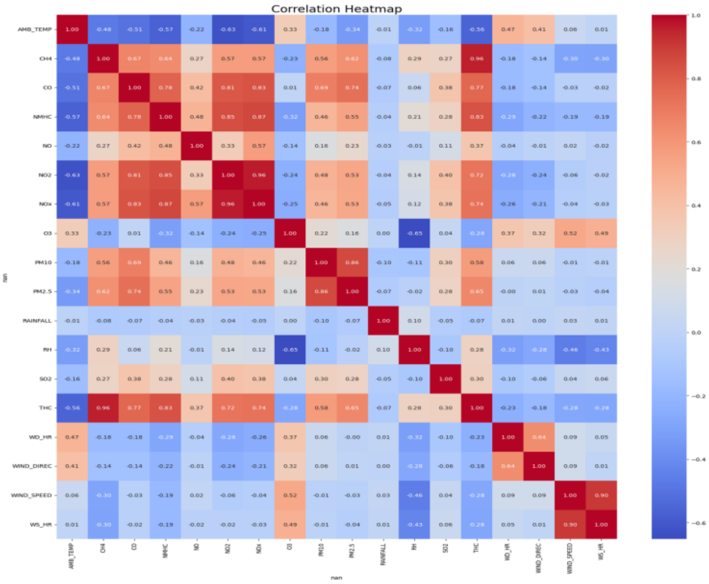


*Figure 6: Co-relation map between numerical features*

The numerical data was available from January to December 2022, but the images were from February 2022 to January 2023. So, we finally train the model on data from February to December 2022.

## 3.3. Model Architecture

This paper presents a novel architecture for estimating PM2.5 and PM10 values using data from two modalities - numerical and images, leveraging GRUs, CNNs, residual connections and self attention mechanisms. The key considerations while developing the model were :

- Capturing temporal relationships in data.
- Efficiently extracting the relevant features from input images
- Model should be lightweight for edge devices
- Should work in a federated setting.

The model follows the fed-multimodal framework. It contains two paths for the numerical and image features before the multimodal fusion step:

- The numerical features are passed through a GRU to capture the temporal dependencies. The current numerical features are also sent forward (short-path) to better represent the current time step.
- The image features are passed through a MobileNetV3 model to extract the 128 dimensional image features.

Multimodal fusion:

- The fusion step has two option – model agnostic feature concatenation, and self attention based fusion. From our initial experiments, the self attention is leading to a decrease in performance.

After the fusion, the combined features are passed through a fully connected neural network. This network outputs 2 numerical values corresponding to PM2.5 and PM10 respectively.
Residual connections were added in this network for the following reasons:

- It helps mitigate the vanishing gradient problem in deep networks.
- It allows the network to learn identity functions easily, which can be beneficial for certain layers.
- It often leads to faster convergence and better performance.
- Increased stability during training due to the layer normalization.

This solved the problem of the testing loss increasing after some iterations. Stabilized the training and overfitting issue of the model.

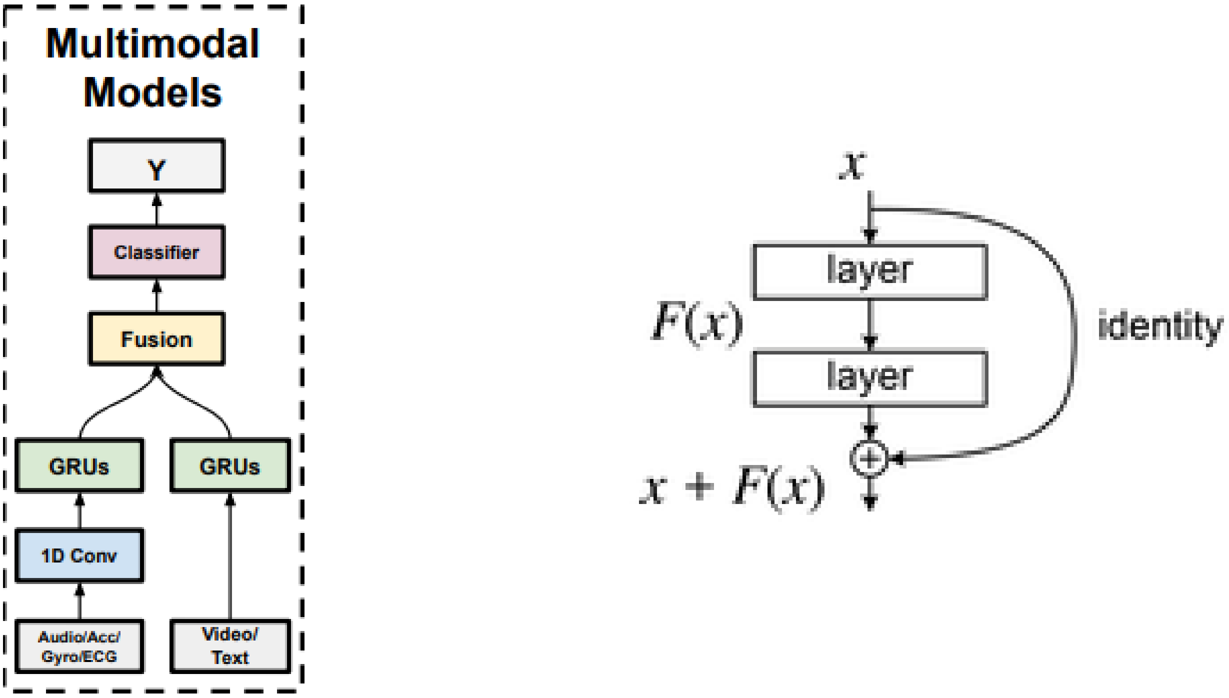


*Figure 36: 1) Fed-Multimodal Framework 2) Residual Connection*

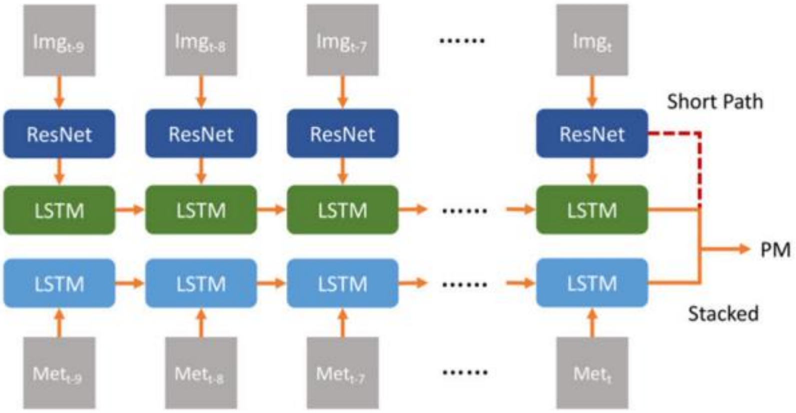
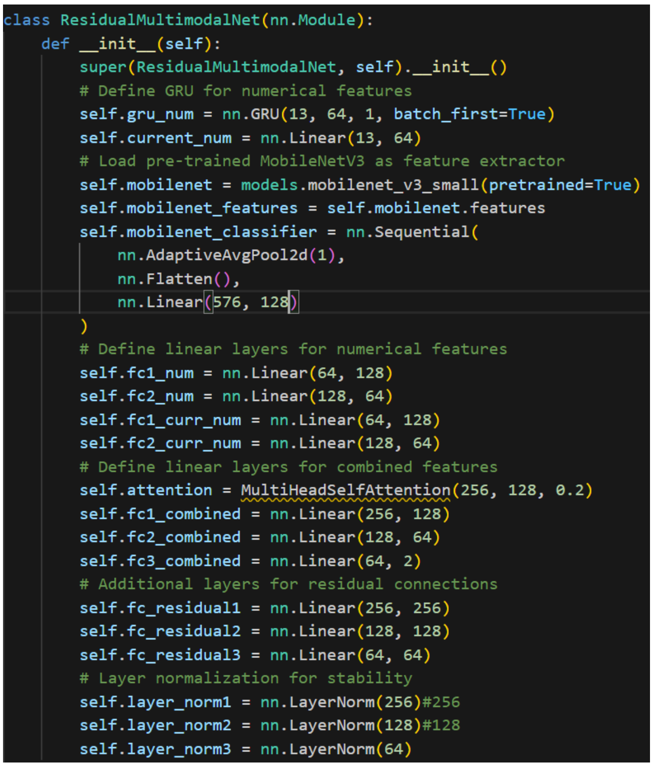
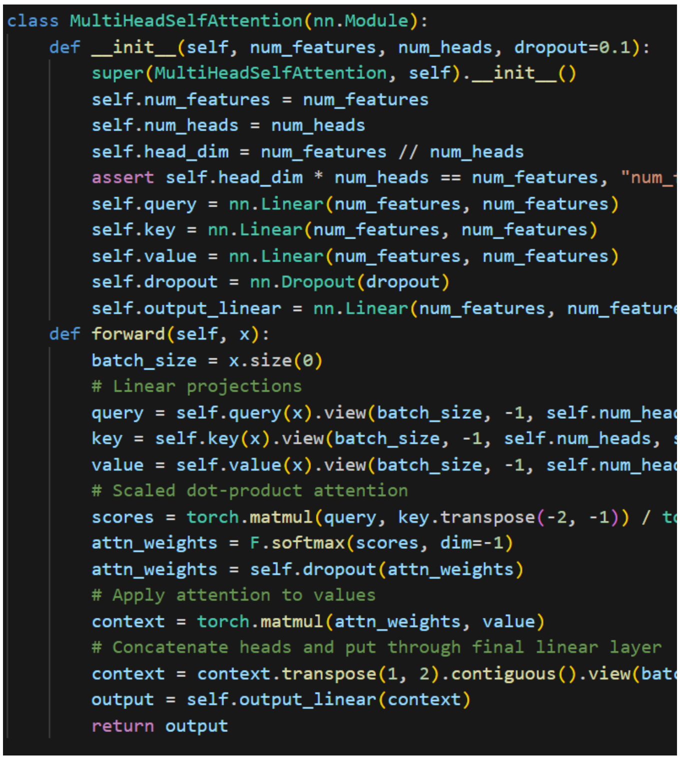


*Figure 37: Short path for current numerical features*



```
class ResidualMultimodalNet(nn.Module):
    def __init__(self):
        super(ResidualMultimodalNet, self).__init__()
        # Define GRU for numerical features
        self.gru_num = nn.GRU(13, 64, 1, batch_first=True)
        self.current_num = nn.Linear(13, 64)
        # Load pre-trained MobileNetV3 as feature extractor
        self.mobilenet = models.mobilenet_v3_small(pretrained=True)
        self.mobilenet_features = self.mobilenet.features
        self.mobilenet_classifier = nn.Sequential(
            nn.AdaptiveAvgPool2d(1),
            nn.Flatten(),
            nn.Linear(576, 128)
        )
        # Define linear layers for numerical features
        self.fc1_num = nn.Linear(64, 128)
        self.fc2_num = nn.Linear(128, 64)
        self.fc1_curr_num = nn.Linear(64, 128)
        self.fc2_curr_num = nn.Linear(128, 64)
        # Define linear layers for combined features
        self.attention = MultiHeadSelfAttention(256, 128, 0.2)
        self.fc1_combined = nn.Linear(256, 128)
        self.fc2_combined = nn.Linear(128, 64)
        self.fc3_combined = nn.Linear(64, 2)
        # Additional layers for residual connections
        self.fc_residual1 = nn.Linear(256, 256)
        self.fc_residual2 = nn.Linear(128, 128)
        self.fc_residual3 = nn.Linear(64, 64)
        # Layer normalization for stability
        self.layer_norm1 = nn.LayerNorm(256)#256
        self.layer_norm2 = nn.LayerNorm(128)#128
        self.layer_norm3 = nn.LayerNorm(64)
```

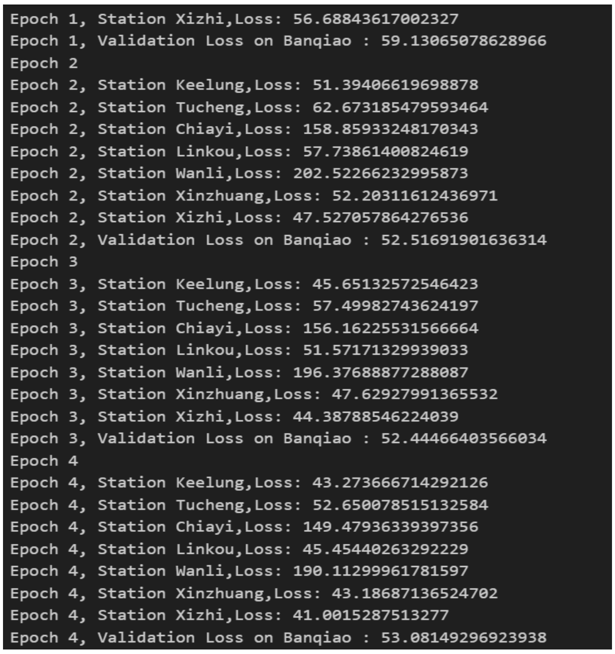


```
class MultiHeadSelfAttention(nn.Module):
    def __init__(self, num_features, num_heads, dropout=0.1):
        super(MultiHeadSelfAttention, self).__init__()
        self.num_features = num_features
        self.num_heads = num_heads
        self.head_dim = num_features // num_heads
        assert self.head_dim * num_heads == num_features, "num_
        self.query = nn.Linear(num_features, num_features)
        self.key = nn.Linear(num_features, num_features)
        self.value = nn.Linear(num_features, num_features)
        self.dropout = nn.Dropout(dropout)
        self.output_linear = nn.Linear(num_features, num_feature
    def forward(self, x):
        batch_size = x.size(0)
        # Linear projections
        query = self.query(x).view(batch_size, -1, self.num_hea
        key = self.key(x).view(batch_size, -1, self.num_heads,
        value = self.value(x).view(batch_size, -1, self.num_hea
        # Scaled dot-product attention
        scores = torch.matmul(query, key.transpose(-2, -1)) / t
        attn_weights = F.softmax(scores, dim=-1)
        attn_weights = self.dropout(attn_weights)
        # Apply attention to values
        context = torch.matmul(attn_weights, value)
        # Concatenate heads and put through final linear layer
        context = context.transpose(1, 2).contiguous().view(bat
        output = self.output_linear(context)
        return output
```

*Figure 38: Model Architecture. The diagram on right shows the self attention mechanism*

## 3.4. Training and Testing

The model was trained on the data for 7 stations : Keelung, Tucheng, Chiayi, Xizhi, Xinzhuang, Linkou, Wanli. The testing dataset was for the station Banqiao.
The numerical input has the shape (batch_size, seq_len, 13) and the image features have the shape (batch_size, 3,224,224).



```
Epoch 1, Station Xizhi,Loss: 56.68843617002327
Epoch 1, Validation Loss on Banqiao : 59.13065078628966
Epoch 2
Epoch 2, Station Keelung,Loss: 51.39406619698878
Epoch 2, Station Tucheng,Loss: 62.673185479593464
Epoch 2, Station Chiayi,Loss: 158.85933248170343
Epoch 2, Station Linkou,Loss: 57.73861400824619
Epoch 2, Station Wanli,Loss: 202.52266232995873
Epoch 2, Station Xinzhuang,Loss: 52.20311612436971
Epoch 2, Station Xizhi,Loss: 47.527057864276536
Epoch 2, Validation Loss on Banqiao : 52.51691901636314
Epoch 3
Epoch 3, Station Keelung,Loss: 45.65132572546423
Epoch 3, Station Tucheng,Loss: 57.49982743624197
Epoch 3, Station Chiayi,Loss: 156.16225531566664
Epoch 3, Station Linkou,Loss: 51.57171329939033
Epoch 3, Station Wanli,Loss: 196.37688877288087
Epoch 3, Station Xinzhuang,Loss: 47.62927991365532
Epoch 3, Station Xizhi,Loss: 44.38788546224039
Epoch 3, Validation Loss on Banqiao : 52.44466403566034
Epoch 4
Epoch 4, Station Keelung,Loss: 43.273666714292126
Epoch 4, Station Tucheng,Loss: 52.650078515132584
Epoch 4, Station Chiayi,Loss: 149.47936339397356
Epoch 4, Station Linkou,Loss: 45.45440263292229
Epoch 4, Station Wanli,Loss: 190.11299961781597
Epoch 4, Station Xinzhuang,Loss: 43.18687136524702
Epoch 4, Station Xizhi,Loss: 41.0015287513277
Epoch 4, Validation Loss on Banqiao : 53.08149296923938
```

*Figure 49: Training and Validation losses*

## 3.5. Results and Analysis

The generated model has achieved good performance , and a good generalization and performance on unseen data.

**Industrial Sector – Coal**

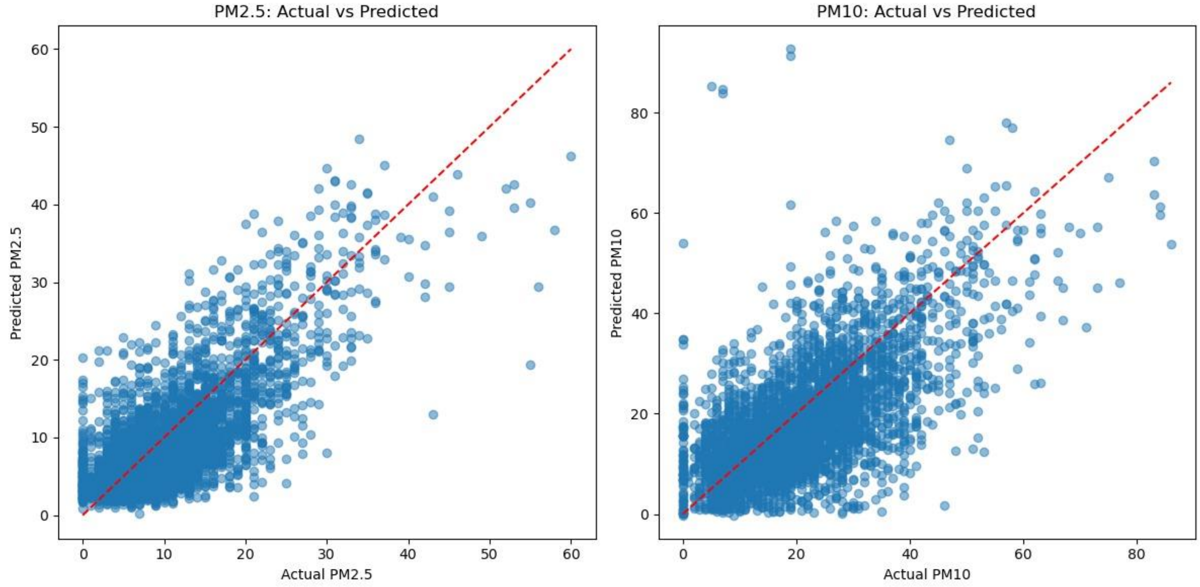


*Figure 56: Industrial Coal Correlation Matrix[12][13][14][18][19][22]*

From this figure, we can see that the model is performing well in general.

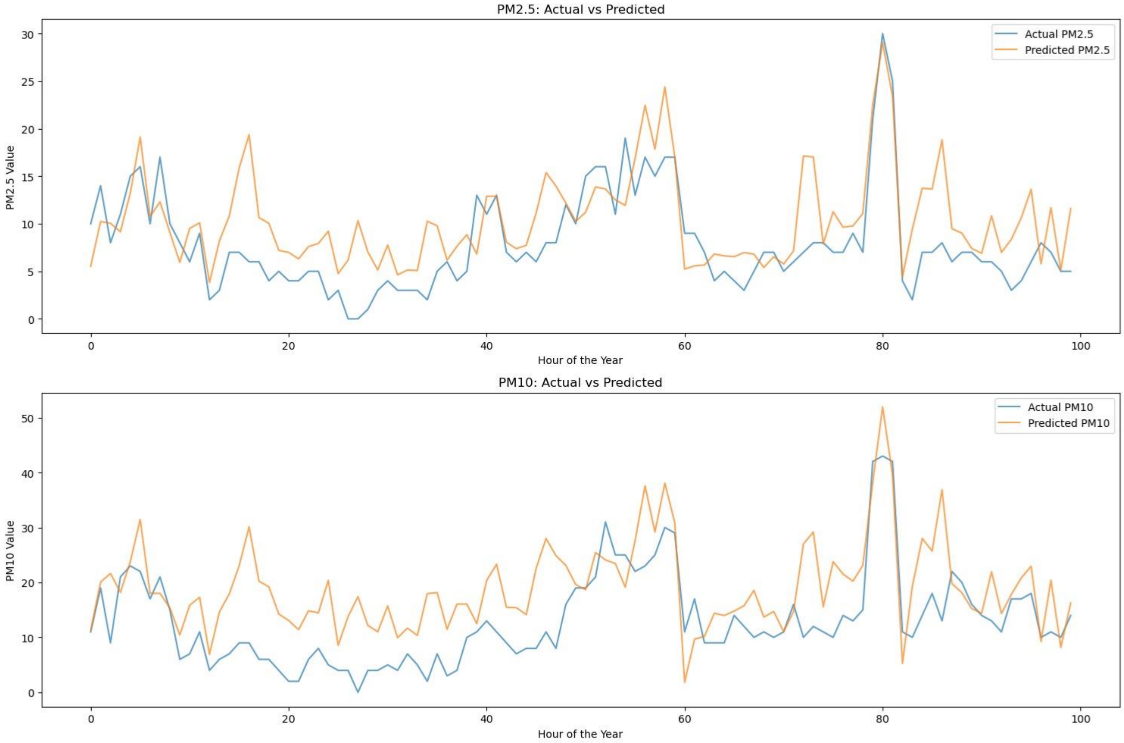


*Figure 57: Plots for the*

From this figure, we can see that the model is struggling to capture sudden variations in pm values in small regions.

**Industrial Sector – Natural Gas**

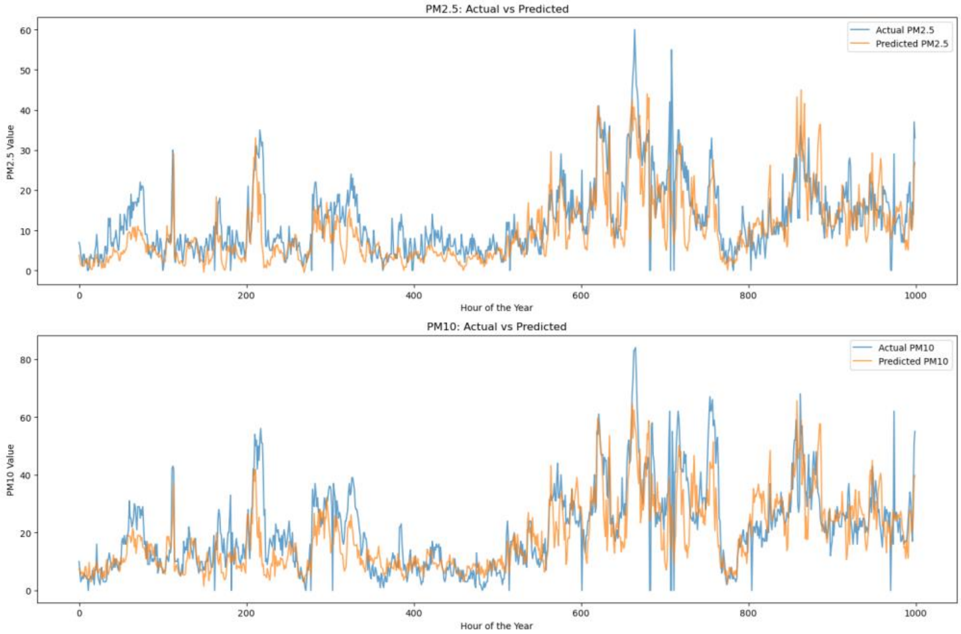


*Figure 58: Plots for the testing dataset over 1000 hours*

The model is able to capture long term variations in PM values successfully.

The losses achieved on testing dataset of Banqiao for the entire year are:
PM2.5 – 5.4
PM10 – 8.8

# 4. Challenges

- The main challenges faced were the data heterogeneity issue. There is a lot of difference between the day and night images,
- Finding an appropriate multimodal fusion strategy.
- Finding and comparing various model architectures.
- The model struggled more with certain stations.
- The variation between pm values throughout the day is a lot.
- Implementing attention mechanisms and residual connections.
- Making a lightweight model that can train on edge devices.
- Handling the missing and corrupted images and numerical data.

# 5. Key Learnings

- The multimodal model performs much better than a unimodal CNN based architecture.
- Kernel based methods for projecting input features to higher dimensional spaces are not viable, as for predictions, the entire training data will be needed , which goes against the principles of federated learning.
- Increased complexity of the model led to a reduction in performance, which was fixed by adding residual connections.
- The self attention based fusion mechanism reduced model performance. Maybe this is because of the model complexity, so need to try with residual connections.
- Giving a sequence to the GRU did not provide much performance benefits. Need to test this extensively.The model is efficiently capturing the long range pm value dynamics, but is struggling with the short term variations.This can be attempted to solve by training a component that predicts the difference between the actual and predicted output. This can be added to generate the final outputs.
- Generating a day night split for the image data does not guarantee better results.
- The reason for the poor performance on the day time images could be because of the colour information. Need to test this out.

# 6. Future Goals

- Understand why the self attention based fusion is not leading to performance improvements. Check if residual connections can address this issue.

- Incorporate a LSTM at the end for predicting the pm values. We can approach this by creating a separate model than predicts the next pm value from previous pm values, and pass the results of our base predictor through this network to get the final outputs.

- Compare encoder-decoder methods for replacing missing images with nearest neighbors.

- Understand why the model is struggling more with pm10 than pm2.5 values. Also why it fails to capture short term variations.

- Enhance model robustness to image transformations by integrating diverse image representations as additional modalities.

- Stack based LSTM Autoencoders showing promising results in one paper. Try implementing it.

# 7. Conclusion

# 8. References

- Fed-Multimodal : https://dl.acm.org/doi/pdf/10.1145/3580305.3599825
- MM Learning : https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10041115
- Residual Networks : https://arxiv.org/pdf/1512.03385
- Unsupervised Pretraining : https://www.nature.com/articles/s41598-019-55320-6.pdf
- MKL: https://jmlr.csail.mit.edu/papers/volume12/gonen11a/gonen11a.pdf
- Transformers: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10153685&tag=1
- PM Estimation : https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9279204
- Spatial and Context Attention Block : https://www.sciencedirect.com/science/article/pii/S0048969720316910?via%3Dihub
- Federated Learning : https://arxiv.org/pdf/2206.12949