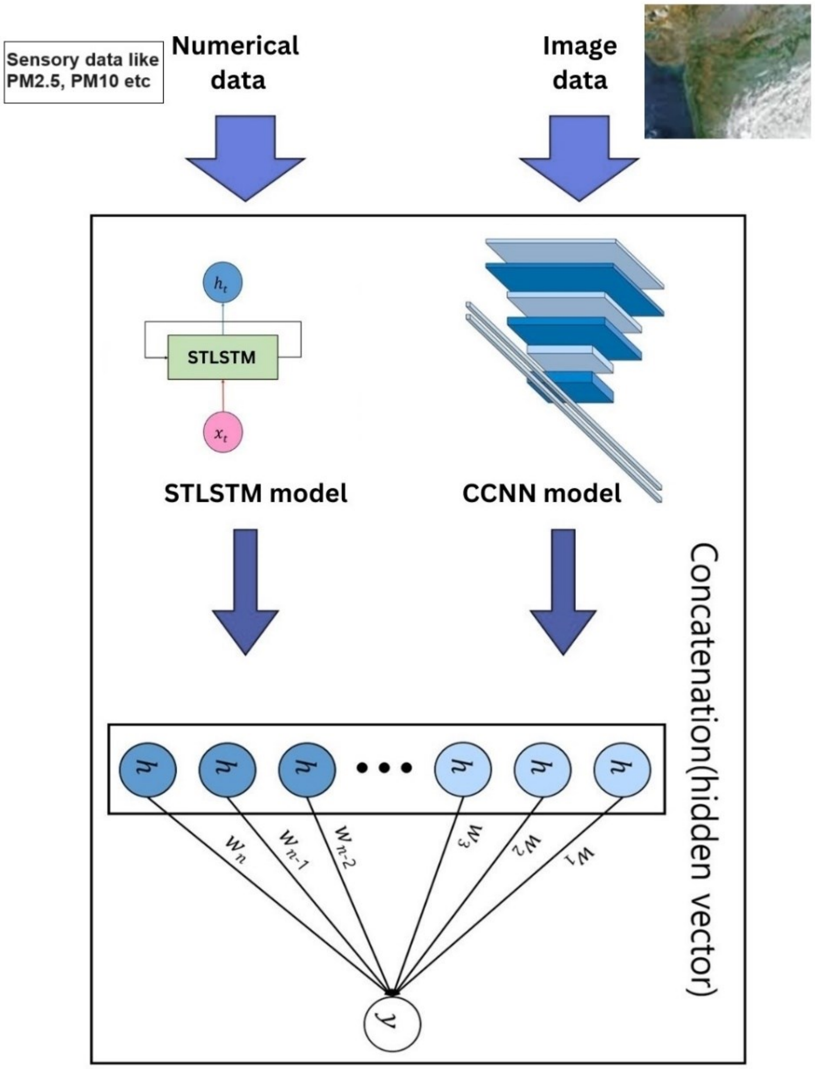
Development of an Air Quality Forecasting System Using a Multimodal Deep Learning Framework Based on Satellite Images of Delhi City

Gowthami S
Associate professor
Department of Electronics and Communication Engineering
Sri Krishna College of Engineering and Technology
Coimbatore
gowthamiiselvarajan@gmail.com

Graphical Abstract



Abstract

With the rapid expansion of science and technology, there is a growing hazard to public health from many forms of pollution in the air, particularly fine dust, which can aggravate or induce heart and lung disorders. Furthermore, this threat is expanding as a consequence of the rapid progression of technology. The purpose of this study is to make an attempt to forecast the fine dust concentration in Delhi eight hours in advance in order to reduce the potential adverse impacts on health. The objective of this study is to develop a multimodal deep learning framework that combines the architectures of Self Tuned Long Short Term Memory (STLSTM) and Concatenated Convolutional Neural Network (CCNN) in order to generate accurate predictions. This research is constructed using a dataset that contains both numerical and visual data. An STLSTM AutoEncoder is responsible for handling numerical time series data, in contrast to the Concatenated Visual Geometry Group Neural Network (CVGGNet) models (CVGG16 and CVGG19), which use image data to compare performance depending on network depth. Based on the results of the final investigation, it has been predicted that the deeper CVGG19 model performs up to 14.2% improved than modality models with single data input that simply use numerical data. The RMSE, MAE, SMAP of the proposed model is 3.87, 3.45, 09.87 respectively. When compared to the models with single data input, the multimodal deep learning model that makes use of both types of data performs significantly better.

Keywords: Air Pollution, Fine Dust, Health Impact, Multimodal Deep Learning, STLSTM AutoEncoder and Concatenated VGGNet.

I.INTRODUCTION

The standard of life has increased as technology and science progress daily, but there is also an increase in air pollution in many forms. Heart and lung disorders are among the conditions that fine dust either causes or exacerbates [1]. Models for prediction of data from time series have been used in numerous research on air pollution to forecast fine dust and avert such health harm. Aerosol data, however, are not included in the majority of research, which instead contain a variety of data types (such as PM-10, temperatures, humid point, and wind speed). Because of atmospheric dispersion, the term "aerosol" refers to fine materials moving in the atmosphere, such as fine dust, and it is helpful for forecasting the movement and buildup of fine dust.

Adding aerosol imaging information that is closely linked with tiny particles to the numerical datasets that performed best in prior study [2] improved the efficacy of the model. The satellite picture such as information on aerosol size distribution was included in the multimodal information set, and it was organised hourly to match the preset numerical information in hourly units. Additionally, the satellite picture offers aerosol data that covers the whole Korean Peninsula, making it possible to see the overall movement of fine particles across the peninsula. In this paper, we present a deep learning model that is a hybrid that combines the LSTM (Long Short Term Memory) series approach, which shows improved precision in time-series information forecasting, with the CNN parallel designs, useful for processing images for learning a particular information set. Processing a high number of features is necessary because the bidirectional deep learning model presented in this study combines the features of the image and numeric datasets that are processed using a CNN series model. Consequently, the LSTM series architecture used the LSTM AutoEncoder, which worked optimally when packed with multiple capabilities. A simple CNN as well as VGGNet (VGG16, VGG19) were utilized by the CNN series model in the prior study [2] to examine the differences as per depth of the network.A multi-layer standard deep CNN architecture, the VGGNet model was created by Oxford University researchers, the Visual Geometry Group.

Therefore, compared to, the multidimensional deep learning models that used both numerical and visual data outperformed it. Using VGG19, the CNN series model with the deepest network depth, produced the best results out of all of them. Furthermore, we separated the image information into original and cropped photos, then ran a multidimensional deep learning model on each to examine the differences in performance. Utilizing katib, an adaptive hyperparameter optimization system, the model's hyperparameter set was tuned to optimal performance. The proposed model makes three notable contributions:

1. The study introduces a novel multimodal deep learning framework that combines Self Tuned Long Short Term Memory (STLSTM) and Concatenated Convolutional Neural Network (CCNN) architectures. This approach is innovative in integrating both numerical and visual data to improve the accuracy of fine dust concentration predictions.
2. By leveraging the strengths of both STLSTM for numerical time series data and CVGGNet (CVGG16 and CVGG19) for image data, the study achieves significant improvements in prediction accuracy. The deeper CVGG19 model, in particular, performs up to 14.2% better than models that use only numerical data.
3. The study's primary aim is to mitigate the adverse health impacts of fine dust pollution by forecasting its concentration in Delhi eight hours in advance. The improved prediction accuracy achieved through the multimodal deep learning model can provide timely warnings and enable better public health responses, thereby reducing the potential health risks associated with air pollution.

The structure of the paper is as follows. The created dataset is presented in Section 3 and associated research is described in Section 2. Subsequently, the suggested framework for the dataset is described in Section 4, and the hyperparameter-based optimizing of the specified model is explained in Section 5. The experiment is described in Section 6, and Section 7 concludes with a discussion of future directions.

II. Related Research

Numerous prediction time series models have been researched for the purpose of weather forecasting. In order to estimate future PM10 levels, [3] used deep learning models with AirNet, which is a time series of weather and pollution data. For data from time series learning in the present research, RNN, LSTM, and GRU were employed; GRU worked best because resource issues limited the usage of AirNet. [4] proposed using Conditions Normalized Models (WNM) based on deep learning to measure variations in air quality in Quito, Ecuador during the COVID-19 partial shutdown period.

Furthermore, [5] introduced an LSTM model that investigates different topologies, including single- and multilayer LSTM, and adds intermediate variable signals to LSTM memory blocks in order to estimate conditions within the Indonesian airport region. The suggested model demonstrated how the addition of the intermediary variable could improve the predictive power of the model.

For the purpose of forecasting PM2.5 levels in Beijing, China, Bekkar [6] presented a hybrid model that combines CNN and LSTM. Based on the combination of models, we apply data from time series for retrieved values using LSTM and apply CNN to extract internal and spatial features for input values. In terms of performance, the proposed model outperformed the current deep learning models (LSTM, Bi-LSTM technology, GRU, and Bi-GRU).The previously stated studies, in contrast to our methodology, lack the use of multimodal data and instead employ time-based prediction models like RNN, LSTM, and GRU.

Few studies use multimodal data to improve the prediction performance of weather forecasting.. A one-dimensional convolution layer of the CNN is used to extract and integrate the spatial correlation characteristics and local variation trends from multimodal air quality data Xie [7]. A GRU uses the CNN results to determine long-term dependencies.The suggested model outperformed the three single deep learning models that are currently in use: Artificial Neural Networks, LSTM, and GRU.

Researcher in [8] recommended using multiple modalities, which consists of picture images collected by cameras in Skopje and meteorological information obtained by sensors, for estimating (i.e., determine if photos obtained with observers are now polluted) by contaminants in the air in the Skopje geographical region in northern Macedonia. In the suggested approach, a new sub-model route joins the previously trained genesis modelling's analysis of picture information, weather information, and all three levels that are linked to the entirety of the layer again. Pretrained inception, CNN, ResNet, and other models were outperformed by it in terms of performance.

Our study is not the same as that of [7] and [8]. While [7] uses both numerical and image information for their multimodal data, we employ both numerical and image data from different places for things like wind speed, SO2, and PM10. Kalajdjieski's art attempts to categorize pictures into contaminatedor clean, making use of sensory meteorological data (temperature, for example). Our objective, in contrast to Kalajdjieski's work, is to forecast future PM10 levels using satellite images and numerical meteorological data.

In order to assemble multimodal data, this work added picture data, which contains information on the size of aerosol particles. Aerosol aids in the prediction of fine dust by indicating the likelihood of fine dust collection and movement owing to atmospheric diffusion.In order to improve performance, we created multimodal data using both numerical and picture data that are relevant to fine dust. Despite significant advancements in time series models for weather forecasting and air quality prediction, several research gaps remain. Most existing studies, such as those by [3], [4], [5], and [6], rely heavily on time-based prediction models like RNN, LSTM, and GRU, focusing primarily on single-modal data. While these models have shown promise in predicting PM10 and PM2.5 levels and enhancing predictive accuracy with intermediate variables or hybrid approaches, they do not incorporate multimodal data, which could provide a more comprehensive understanding of air quality dynamics. For instance, [7] demonstrated the potential of multimodal data by integrating one-dimensional CNN layers to capture spatial correlations from air quality data, yet this approach still lacks the depth of integrating diverse data sources. Furthermore, Kalajdjieski [8] utilized multimodal data combining images and meteorological information for pollution estimation, but their focus was on current pollution status rather than future predictions. Our study aims to bridge these gaps by employing both numerical and image data from various sources, including satellite images, to forecast future PM10 levels. This approach not only enhances the predictive performance by utilizing aerosol particle size information to indicate fine dust movement but also provides a robust framework for integrating multimodal data relevant to fine dust prediction. Thus, while previous research has laid the groundwork for air quality forecasting, the incorporation of diverse data modalities remains an underexplored area that our study seeks to address, offering a more holistic and accurate prediction model.

III.Dataset

In this study the open source dataset of https://www.kaggle.com/datasets/deepaksirohiwal/delhi-air-quality has been utilized. This dataset comprises air quality data from Delhi, the national capital of India. It includes measurements of various pollutants, such as particulate matter (PM2.5 and PM10), nitrogen dioxide (NO2), sulfur dioxide (SO2), carbon dioxide (CO2), ozone (O3), and others.

The data was collected from monitoring stations across different areas of Delhi over a period from November 25, 2020, to January 24, 2023. This work used image as well as numerical data to anticipate fine dust beyond 8 hours, and the greatest results were achieved when input values dating back up to 5 hours were employed. To further increase performance, we combine equivalent numerical dataset from [2] with satellite image data to create a multimodal dataset in this study.

3.1 Satellite picture Database

Because aerosol information [9] can be used to determine movement and accumulation in accordance with atmospheric diffusion, it was used in this work to predict fine dust. As a result, we created a multimodal dataset using hourly satellite photos with information on aerosol particle size and numerical data. You can notice the general movement of fine dust over the Korean Peninsula by examining the satellite image, which includes aerosol data for the whole peninsula.

An instance of an image from satellite with a color caption for aerosol particle size located in the bottom right corner is presented in Figure 1. The size of aerosol particle is represented by α. The range of α is −0.5–3, and particles size is exponential applying the ratio of each wavelength to the appropriate optical thickness [10]. Figure 1 illustrates how the colors change according on the range of α that is calculated. The blue sequence (α: 0–1) and purple sequence (α: −0.5–0) correspond to big aerosols like sea salt particles and yellow dust, respectively. The yellow and red series (α: 2-3) correlate to fine particles, such as smoke or pollution, whereas the green segment (α: 1-2) denotes medium-sized aerosols.

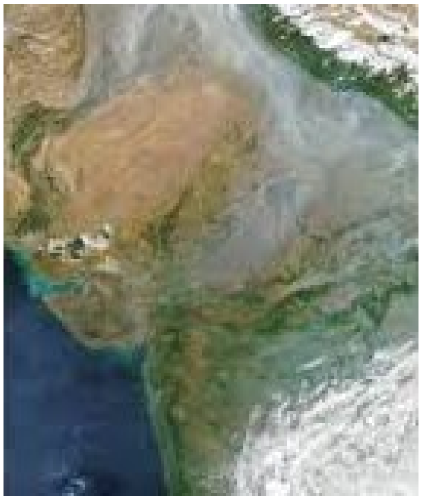


Figure 1. Satellite picture of Delhi

3.2 Cropped Satellite Picture Database

Cropping is the process of deleting portions of an image that are not wanted. The aerosol information for Korea along with other nations is shown in Figure 1. Nevertheless, as the numerical information generated is limited to the Delhi data, the remaining region in the satellite image is eliminated, as seen in Figure 2. The variation in prediction performance based on the satellite image datasets is then demonstrated.

Figure 2. Cropped from the satellite image for further processing

IV. Application of Deep Learning

Depended on STLSTMs & CCNNs models, we build multimodal deep learning algorithms in this work. For historical data, LSTM constitutes a recognized DL, and for picture information, the CNNs is a well-known approach. For our multimodal deep learning approach, which handles both numerical and picture input, we mix the two models.

4.1 AutoEncoder for STLSTM

An issue with RNN [11] is that it cannot transfer historical data all the way through a large time series structure. This is resolved by the LSTM [11] model. There are other LSTM model variations, including bidirectional-LSTM.

This study presents a multimodal deep learning system that combines CNN-treated visual information with LSTM-treated numeric information. More characteristics follow, which results in more dimensions and more challenging acquisition. To address this, we use the LSTM AutoEncoder, that efficiently encrypts the high-dimensional input. In our case, we have already demonstrated that LSTM AutoEncoder performs better than the vanilla LSTM model [2].

A layer on top of the LSTM layers in the LSTM AutoEncoder is used to extract characteristics by lowering the dimension of the input data [12] and producing original data using the features that are extracted. The AutoEncoder is commonly utilized for unsupervised or self-supervised learning scenarios in which the input and label are identical. However, in our earlier research [2], we employed the AutoEncoder for supervised learning. The LSTM AutoEncoder used in our earlier work [2] is seen in Figure 3. The decoder and encoder process the initial input values of the built model, which is data from the last five hours, to forecast PM10 values one to eight hours from now.

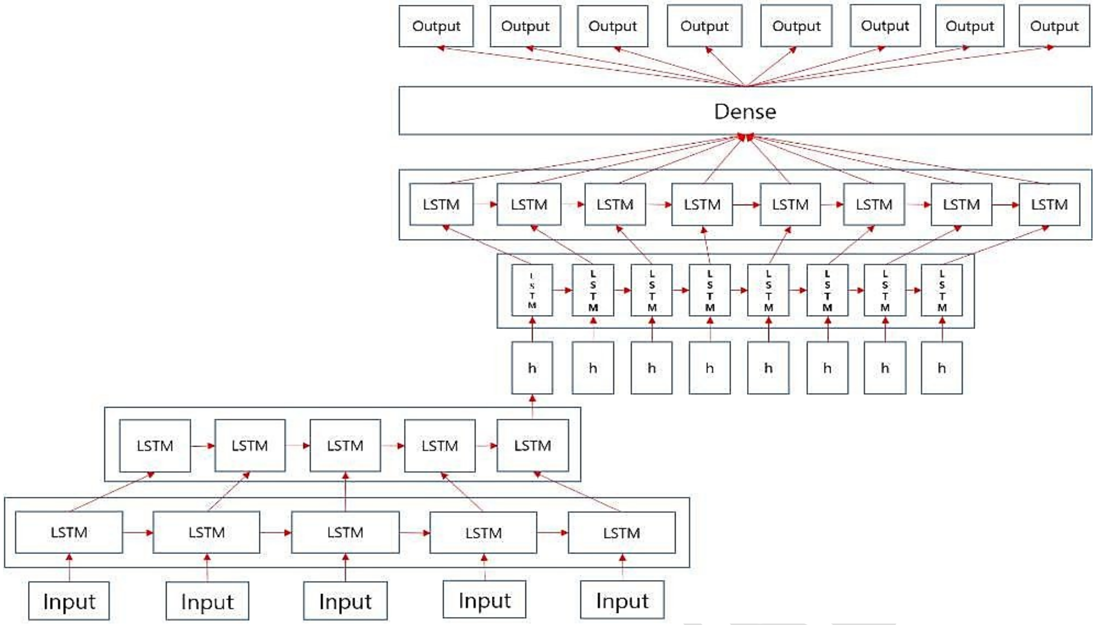


Figure 3. Demonstrates the efficiency of the LSTM AE algorithms in analysing the input information, which is a 5-hour series of values, and generates a total minimum dust value within 1-6 hours

4.2 Working of CCNNs

First, we process satellite pictures using the fundamental CNN [13] in order to examine how well advanced CNN models, like VGGNet, perform. By processing a portion of the image rather than the full one, the CNN model is able to handle the problem of typical neural networks receiving inputs without the need for spatial or topological information. By doing this, the CNN is able to learn images while preserving their spatial information.Convolution layers & pooling layers are so frequently employed in image processing to extract features from the image.Figure 4 depicts the basic CNN that was experimented extensively in this paper. It consists of two fully linked layers and three convolution/pooling layers.

4.3 VGGNet

Assess the efficacy of various CNNs, we experimented utilizing the VGGNet [14], which is a more complex CNN than the basic CNN model. A CNN series model called VGGNet was created by the Visual Geometric Group, aOxford University research group. It is a multi-layered, conventional deep CNN architecture. We selected the VGGNet model due to practical challenges in experiments caused by other models, including the ResNet deep than VGGNet, requiring more storage space. Compared to the standard CNN model, the VGGNet model has a deeper model with a fixed filter size as tiny as 3 × 3.

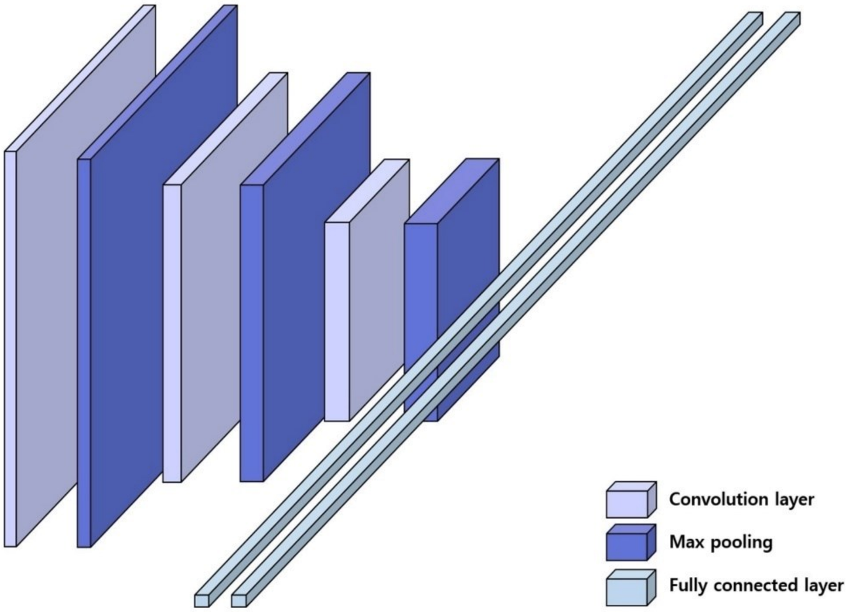


Figure 4. CNN model without pooling layers

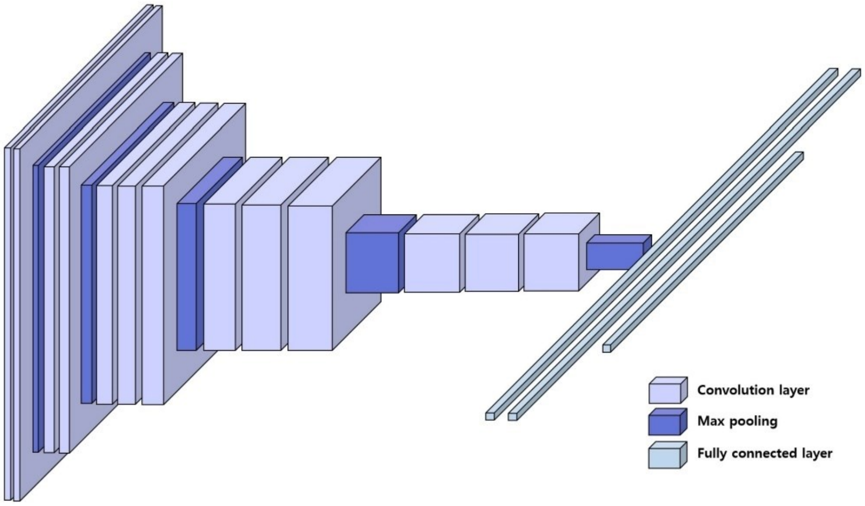


Figure 5. The VGG 16 models without pooling layers

Rather than using a single convolution layer with large-sized filters, VGGNet uses many convolution layers with small in size 3 × 3 filters. Furthermore, because the model can describe high-dimensional nonlinearity, performance improves as the model gets deeper. VGGNet is referred to as VGG16 in Figure 5 Six structures in all—A (11), A-LRN (11), B (13), C (16), D (16), and E (19)—were created by the VGG research team, and their performances were compared to see if there was a difference in performance based on depth. Consequently, it was verified that as the depth grew from 11, 13, 16, to 19 layers, the performance got better. Satellite photos, as well as VGG16 and VGG19 in this paper, are processed using VGGNet.

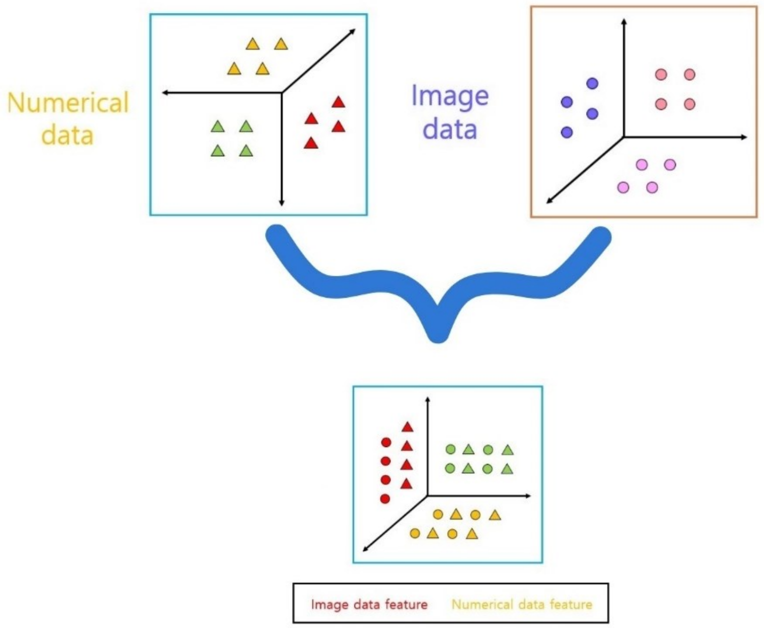


Figure 6. Data having different properties (numerical and image data) are embedded, extracted, and projected onto a common feature space

4.4 Multimodal Deep Learning

A combination of several methods, such text and picture, textual and sounds, or numerical information and picture, is called multimodal. The technique of creating and obtaining the deep learning approach is known as "multimodal deep learning" [15]. We can correlate relationships between several modalities and address a variety of problems via multimodal deep learning. In this research, we build a multidimensional deep learning algorithm using numerical and picture data to learn various data associations based on time series features. Nevertheless, we must integrate these data because picture and numerical data have distinct properties.

Numerous techniques exist for integrating data [16]. First, as As shown in Figure 6, information dimensions may be used to embed information with various properties and retrieve data with similar attributes. Consequently, information with different characterizations are directed towards a same range of features. Second, integration among learned represents is a method for combining different neural network algorithms with learnt visualisations, as seen in Figure 7.We apply both the integration between learnt representations and the integration with the data dimensions in this paper. For instance, we use LSTM for series of data and CNN for picture data during training. Each neural network's learned representations are linked and integrated to the component that is hidden, and learning is accomplished by determining the ideal weight associated with the hidden layer.

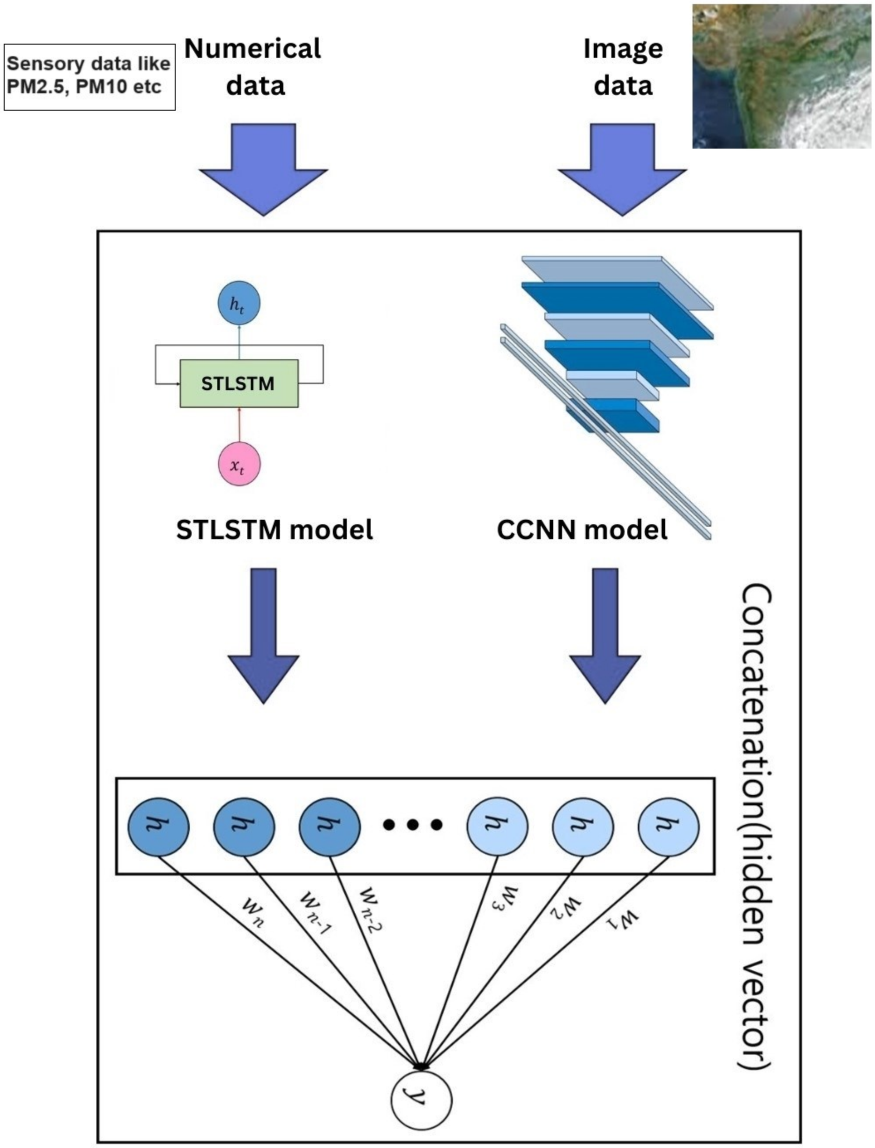


Figure 7. Overall flow of the model

The multimodal DL framework that we built in this paper is depicted in Figure 8. For the purpose of using data from the previous five hours as input, the numerical & image data must first be preprocessed. Even if they are the same, the two dimensions of data do not match. Preprocessing takes place during the five hours before. In order to maintain the spatial details of the image, we recover the feature mapping for picture information using a CNN-based techniques.

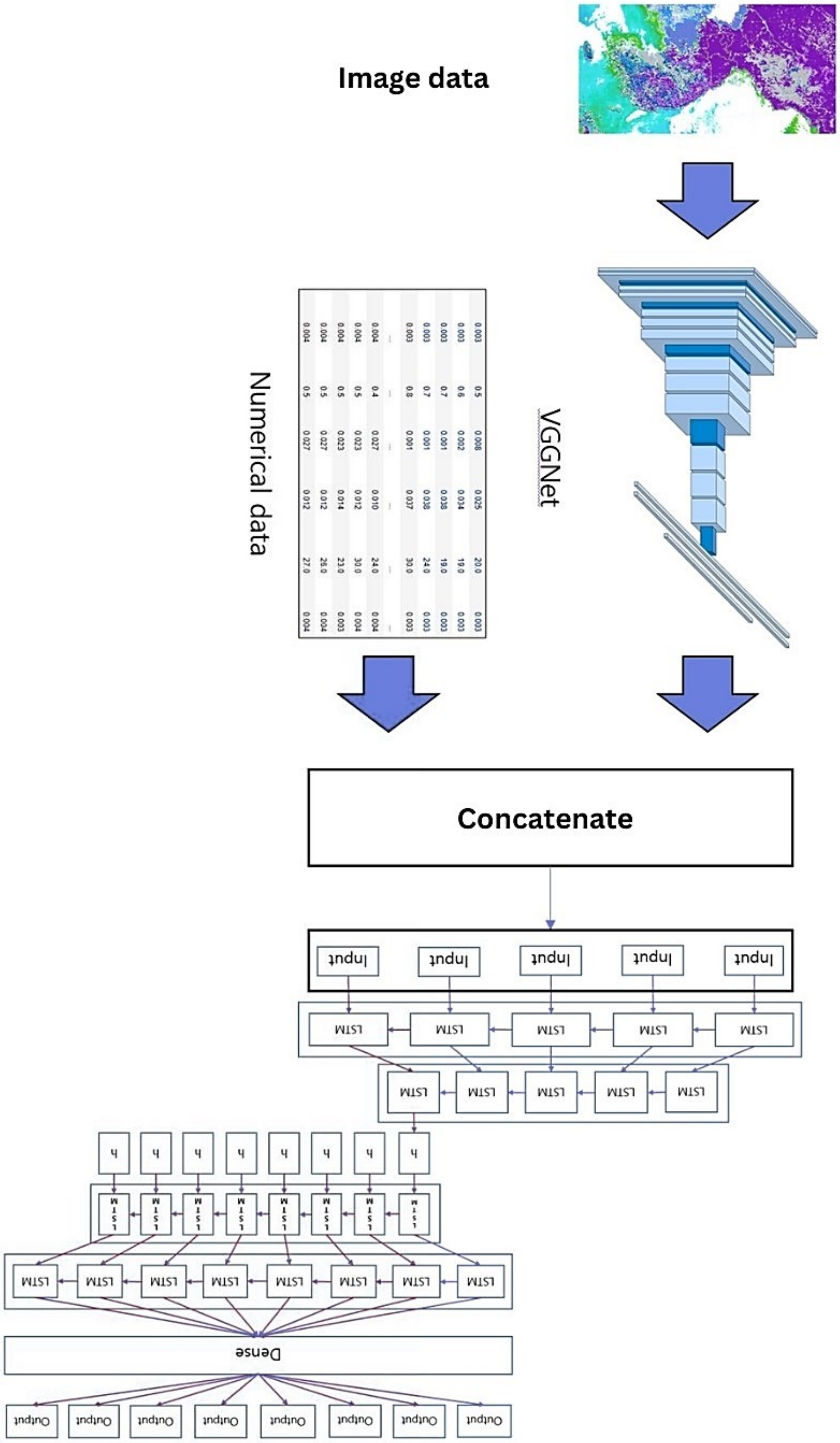


Figure 8: DL Techniques with multiple modes: Using CCNN, the feature mapping of the picture information is first extraction. After that, a concatenate layer is used to integrate

the feature map together numerical data, giving them time series properties then after the combined data is handled by the STLSTMs AE

An LSTM AutoEncoder techniques was employed to process numerical data. Since our parameters rise when numerical and picture data are combined, the GRU framework [17], another efficient time series information handles paradigm, is disregarded. The LSTM model performs better in this scenario than the GRU model.

Furthermore, to compare efficiency differences depending to networking depths, we employed their fundamental CNN & VGGNet models (VGG16, VGG19). The data are combined, nevertheless, in order to potentially increase the dimension and decrease the quality of the learning. We send the integrated multimodal data to our model's LSTM AutoEncoder to indicate the dimensionality after the multimodal data have been integrated.
reduction & time series properties throughout the preparatory phase.

Our multimodal deep learning algorithm differs from the existing multimodal deep learning algorithm used in earlier works in that it makes use of a prediction model for time-series information in order to facilitate long-term dependency learning. The difficulty of classifying photographs as polluted or not, especially when combined with weather data, is addressed by Kalajdjieski [8].

V.Optimizing Deep Learning Models

This research optimises this hyperparameter to maximise the accuracy of the model.

5.1.High-performance parameter

A hyperparameter [13] is a value that the user has to actively set in order to use theprototype. For instance, batch size, epoch, and learning rate. Hyperparameter tuning is the process of examining the best hyperparameters to enhance accuracy, and determining the best values for this is never simple. The ideal value can be found by manually substituting the value, however this may consumed time and you might not obtain the optimized value if you start with the wrong criterion. As a result, in order to determine the ideal hyperparameter, the relevant search method must be chosen based on the implementing scenario.

5.2.Katib

Katib [18] is a system that is part of the Kubeflow ML techniques that optimizes hyperparameters. Katib maximizes accuracy by utilizing many algorithms. Random search, search via grid, & Bayesian optimization are examples of common algorithms. Katib is designed to automate the hyperparameter tuning process in machine learning workflows. It efficiently manages experiments to optimize model performance by exploring various hyperparameter configurations.

Among other techniques, we employed the random search method [19]. When it is not possible to investigate every option, random search is a great strategy to apply since it generates a mixture of random parameters. We experimented with a range of hyperparameters under different circumstances. however, the number of search instances increases significantly if the parameter's range contains decimal points.

VI.Trial

Every experiment pertaining to this suggested framework is conducted on a PC with an Intel(R) Core (TM) i3-8130U CPU running at 2.20 GHz, 64-bit OS x64, and 4.00 GB of RAM. All of the experiments are carried out using the Python programming languages.

6.1.Experimental Configuration

The objective of this work is to use the S_Ns' pollutant content and climatic parameters to estimate the AQI for S_L one hour in advance. To be more precise, the data from the preceding 24 hrs are utilized to forecast their AQIs for the 25th hr. In order to conduct this test, 26,280 information points are used.Of the database, 21,024 information, or 80% of the total, are used for training, while 5,256 data samples, or the remaining 20%, are utilized for testing. An essential first step toward a faultless and effective experiment is data normalization. Data normalization is crucial because it eliminates redundant data and, to some extent, various kinds of irregularities from the data. The time-series data fluctuates across a large range and is quite volatile. The entire process of learning slows down as a result. The data normalization procedure is used to expedite learning and scale the information between zero and one. The Min-Max technique is applied in this case. The data are transformed linearly using this normalization method.This involves taking the dataset's minimum and maximum values and replacing them with the following formula:

$$\boldsymbol{n}_{\text{norm}} = \frac{((\text{ high-low }) * (\text{ n-MinN }))}{(\text{MaxN} - \text{MinN})} \qquad (1)$$

where the lowest and highest numbers of characteristic N in the input dataset are represented by MinN and MaxN, respectively.

Equation 1 is used to transform the input value n, which is an attribute of N, to $\boldsymbol{n}_{\text{norm}}$. Their parameters for the suggested hybrid forecasting model are shown in Table 1.

Table 1: The suggested hybrid prediction model's parameter description

CNN_Block				GRU_Block				Epoch
Filter_size			Dropout	No. of neurons			Dropout	
Layer 1	Layer 2	Layer 3		Layer 1	Layer 2	Layer 3		
73	73	73	0.7	17	17	17	0.7	5-151
73	73	73	0.7	33	33	33	0.7	5-151
73	73	73	0.7	65	65	65	0.7	5-151
73	73	73	0.7	81	81	81	0.7	5-151
73	73	73	0.7	129	129	129	0.7	5-151

6.2.Assessment

This section evaluates the suggested prediction model and the imputation procedure for filling in the dataset's missing values in order to determine their suitability.

6.2.1. Assessment of the suggested imputation technique for the replacement of missing values

Numerous imputation algorithms are now in use. They are frequently employed in dataset replacement for missing values. Average Imputing [19], An autoregressive , also [20], the highest probability [21], K-NN, also [22], and Bagging the following section [23] are a few of the well-known attribution algorithms [24,25]. The suggested imputation algorithm is compared to the current imputing techniques in order to assess its applicability. Table 2 presents the findings.

6.2.2.The prediction model's assessment

It is crucial to assess a model's performance once it has been built. All regression models, regardless of kind, need to go through an evaluation process in order to filter the model's errors and compare the suggested model's performance to that of the established models. Here, three assessment measures are applied to assess the hybrid prediction model's effectiveness. They are listed below.

The Mean Absolute Error (MAE) : is a commonly used error evaluation technique for predicting the mistakes in a time-series study. The procedure to ascertain loss of function MAE is applied during training. The MAE is better able to depict the real-world errors that were made during the prediction framework's training phase. The formula for MAE is

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} |X_i - Y_i| \qquad (2)$$

RMSE, is a useful tool for analyzing the discrepancy between the observed and anticipated values. A reduced RMSE score is always indicative of a high-quality prediction model. RSME is then computed as

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (X_i - Y_i)^2} \qquad (3)$$

Symmetric Mean Absolute Percentage Error, (SMAPE), is an error evaluation technique.Thus, SMAPE is defined by applying the subsequent formula:

$$\text{SMAPE} = \frac{100\%}{N}\sum_{i=1}^{N} \frac{|X_i - Y_i|}{(|X_i| + |Y_i|)/2} \qquad (4)$$

where i is the number of observations, Y_i is the actual value, X_i is the predicted value. The efficiency of our suggested work is assessed by contrasting the same framework with a few common models, such as:

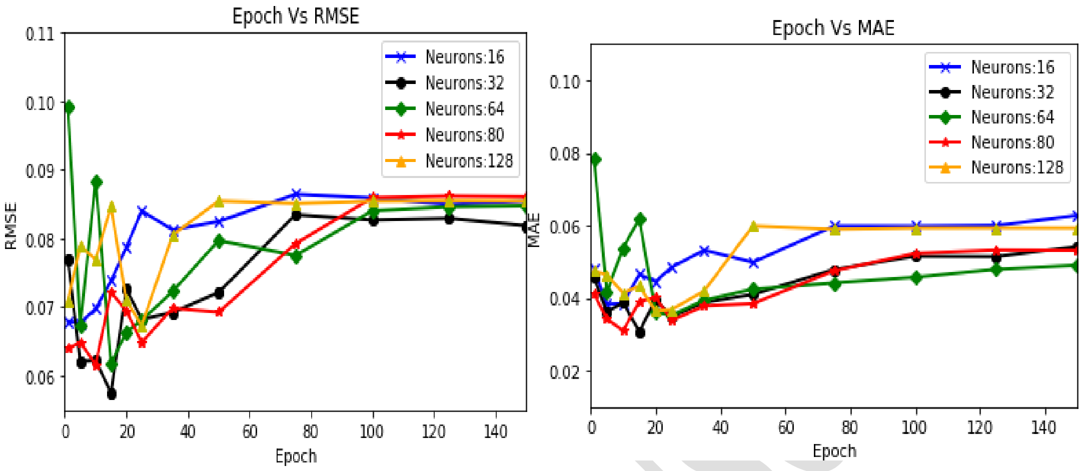
- Support Vector Regressor, or SVR
- Stacked LSTM: The forward as well as backward time-series data are analyzed using three Bi-Directional LSTM layers.
- GRU: Bi-GRU is applied in three layers.
- CBGRU
- DAQFF

VII.Findings and Discussion

7.1.STLSTM-CCNN Simulation

STLSTM-CCNN were simulates using the factors shown in Table 1 in order to obtain the best outcome possible for the suggested STLSTM-CCNN framework. In fig 9 From there, it is evident that, with 32 neurons and a 16-epoch number, STLSTM-CCNN is producing prediction results that are more accurate. Here, the drop-out layers of the CNN & GRU blocks are both employed with a rate of dropout of 0.6 in order to get around the overfitting issue. These parameters are applied to the remaining experiments.

The STLSTM-CCNN simulates with different wind speed threshold values for the adjacent Place's Impact (NPI) method in order to determine the optimal thresholds value for the wind speeds (ϕ) for each of the adjacent station. Based on the results of the studies, the threshold value ϕ has been allocated to the following stations: 1.74 m/s for Richmond, 2.05 m/s for St. Marry, 1.77 m/s for Bringelly, 1.93 m/s for Liverpool, and 1.76 m/s for Randwick.



(a)RMSEs. *(b) MAEs.*

Figure 9. Analysis of performance (a) RMSEs (b) MAEs.

7.2.Effectiveness of suggested imputation techniques for replaced missed values

The efficacy of the suggested imputation algorithm in substituting absent values The univariate time-series dataset has 5%, 10%, 15%, 20%, and 30% of its data randomly removed in order to test the efficacy of the suggested imputation algorithm. The missing values are then replaced using the tried-and-true imputation procedures. The RMSE and MAE are computed based on the predictions made by the prediction model. The performance of different imputation techniques on the deletion of the aforementioned data segments from the dataset is shown in Table 2. This table shows that all of the imputation algorithms' errors grow as the proportion of missing values rises. Additionally, compared to other imputation methods that have a rising percentage of missing values, the rise in predictions error rates is substantially smaller. In contrast, the Mean/Mode impute algorithm performs the worst and increases error rates. With the exception of the bagging technique, it is evident that the suggested seasonality-based imputation approach yields the least amount of error. The Bagging algorithm outperforms the suggested Seasonality-based imputation approach by a little margin their losing percentage is larger or equal to 15%.

However, it can also be seen that its RMSE as well as MAE are marginally higher than those of the Bagging method alone when the missing amount is more than or equal to 15%. It is clear from this debate that missing values more than 10% can likewise be accommodated by the

Seasonality-based imputation technique suggested in this letter. Less than 10% of the dataset utilized for the proposed work are missing values. Because the suggested imputation algorithm learns seasonal patterns and manages the missing information appropriately, it performs better.

Table 2: An assessment of the suggested imputation techniques performance

Techniques	% of missing ranges									
	5%		10%		15%		20%		30%	
	RMSEs	MAEs	RMSEs	MAEs	RMSEs	MAEs	RMSEs	MAEs	RMSEs	MAEs
Average	4.56	4.09	5.63	6.99	7.30	7.87	8.35	8.76	8.97	9.34
Autoregressive	2.82	2.45	3.36	3.01	3.80	3.98	4.91	5.12	5.74	7.22
Maximum_likelihood	3.1471	2.9613	3.4206	3.4876	4.0019	4.7204	5.4811	6.8872	6.2553	7.1145
K-NNs	4.22	3.97	4.33	4.67	4.98	6.54	5.67	7.68	7.98	8.76
Bagging	3.81	3.42	4.34	3.98	4.65	5.67	6.76	7.89	6.78	5.87
Proposed	**3.72**	**3.54**	**3.76**	**3.87**	**3.87**	**4.76**	**5.67**	**6.78**	**6.78**	**6.54**

7.3.Performance assessment taking surrounding areas' air quality into account

Two phases of experiments are conducted to investigate the possibility that the environmental condition of a given location may be influenced by that of its neighbors.There are two phases:

- The pollutant amount of S_Ns is taken into account when predicting the AQI of S_L.
- Without taking into account the pollutant content of S_Ns, the AQI of S_L is estimated.

Table 3 demonstrates that the experiment performs better in estimating the air quality index (AQI) of S_L when the amount of pollutants associated with S_Ns are taken into account. However, it is evident that when the impact of S_Ns' pollutant concentrations is disregarded in the diagnosis of S_Ls' air health, the mistake rates increase.

Table 3 compares performance both with and without taking into account the influence of nearby locations

Parameters	**RMSEs**	**MAEs**	**SMAPEs**
Places for neighbor (considered)	2.9013	2.2730	8.9340
Place for neighbor (Ignoring)	3.7621	2.8932	10.8762

7.4.Comparative study with other models

All of the current models, including SVR, BiLSTM, Bi-GRU, CBGRU, and DAQFF, have their RMSE, MAE, and SMAPE values computed in order to assess the correctness of the findings produced by our suggested model, STLSTM-CCNN. The performance assessment of the suggested model, as well as Bi-LSTMs technology, SVRs, Bi-GRUs, CB-GRUs, & DAQFFs, is displayed in Table 4.

It is evident that out of all the models, the suggested model, STLSTM-CCNN, makes the fewest mistakes. In order to assess the prediction models more critically, a compared is made with the same Database from Australia's New South Wales, and the resulting performance graphs are shown in Figure 10.Plotting of the expected versus real AQI is done there. The outcome is magnified here for the 1000 information points. For every model, the data point is displayed from 02/07/2019 at 01:00 hours till 12/08/2019 at 16:00 hours.It is evident from the graphical representations of all three conventional shallow forecasting models (Figures 10a, 10b, and 10c) outperform their conventional MLs Techniques SVR. This is because the three graphs show that the wave peaks and wave valleys of LSTM and GRU agree more frequently than those of SVR. Thus, it may be said that SVR is a shallow machine learning model, whereas LSTM and GRU are shallow deep learning models.

Upon consulting Table 4, it is evident that GRU consistently exhibits the lowest error when compared to LSTM and SVR. Figure 10 presents a comparison between the shallow deeper techniques architecture and their integration of many DLs architectures. Figure 10 makes it evident that CBGRU, a hybrid of 1D-CNNs and Bi-GRUs, and DAQFFs, a hybrid of LSTMs and 1D-CNN, exhibit superior accuracy in comparison to the DLs framework GRU. It is evident by looking at Figure 10a, 10b, 10c, 10d, 10e, &10f that the suggested model, STLSTM-CCNN, is doing better than the other models. The agreement between the wave peaks along with wave troughs between the actual and anticipated values is superior to that of any other model illustrated. Figure 11 shows the boxplot deviation analysis to illustrate their efficiency of their different forecasting techniques.

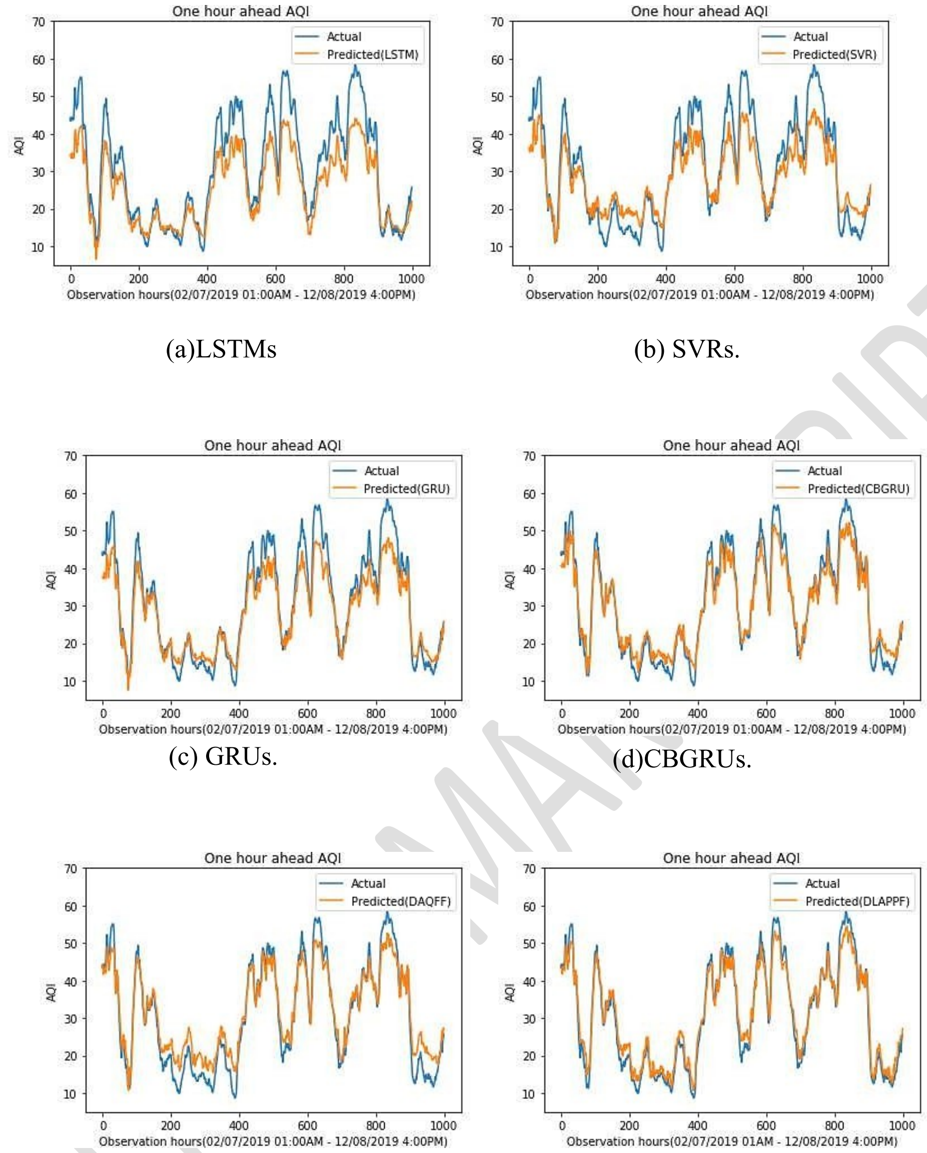
Measured deviation is the discrepancy between the actual and expected results. Shorter whiskers and thicker boxes are used to depict more centralized data. Plotting makes it evident that the suggested models, CBGRU and STLSTM-CCNN, have medians close to zero and notches close to zero. When focusing on this boxplot, it is evident that the STLSTM-CCNN forecasting model has the fattest boxes with the smallest whiskers out of all the models depicted in Figure 11. Therefore, based on this finding, it can be said that when it comes to predicting a place's AQI, STLSTM-CCNN has the highest accuracy.

The merging of many DL architectures in the proposed model's AQI prediction is the key component that makes it successful.There are two main reasons why STLSTM-CCNN is performing better than all other models:

- In order to estimate the AQI of S_L, STLSTM-CCNN can take into account not only the air pollution intensity elements of S_L, but also the PM2.5 along with the additional meteorological variables of the neighboring S_Ns.
- The sequence of CNN layers functions as a building block that enables the model to retrieve increasingly significant local complicated information from the input windows. Furthermore, STLSTM-CCNN makes advantage of the GRU block.The GRU block is made up of three stacks of Bi-GRU, each of which can abstract a temporal feature and the time-series data's backward and forward (both ways)dependencies.

Table 4: Comparison of various Techniques performances

Techniques	**RMSEs**	**MAEs**	**SMAPEs**
LSTMs	7.12	6.76	19.32
SVRs	7.24	6.87	18.98
GRUs	5.87	4.84	13.70
CBGRUs	4.54	3.56	11.54
DAQFFs	4.87	4.54	14.76
STLSTM-CCNNs	3.87	3.45	09.87



(a)LSTMs

(b) SVRs.

(c) GRUs.

(d)CBGRUs.

(e)DAQFFs

(f)STLSTM-CCNNs

Figure 10. Efficiency graph for LSTMs, SVRs, GRUs, CBGRUs, DAQFFs, STLSTM-CCNNs, and GRUs.

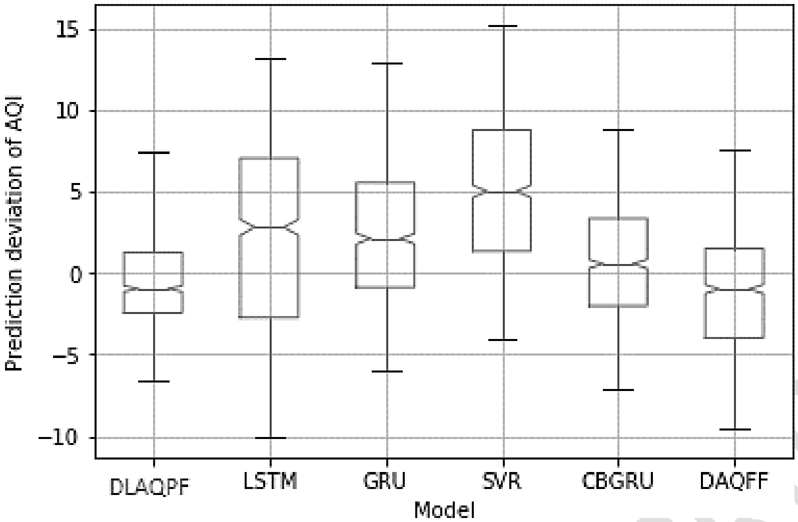


Figure 11: A boxplot that shows how different forecasting models' predictions differ from one another

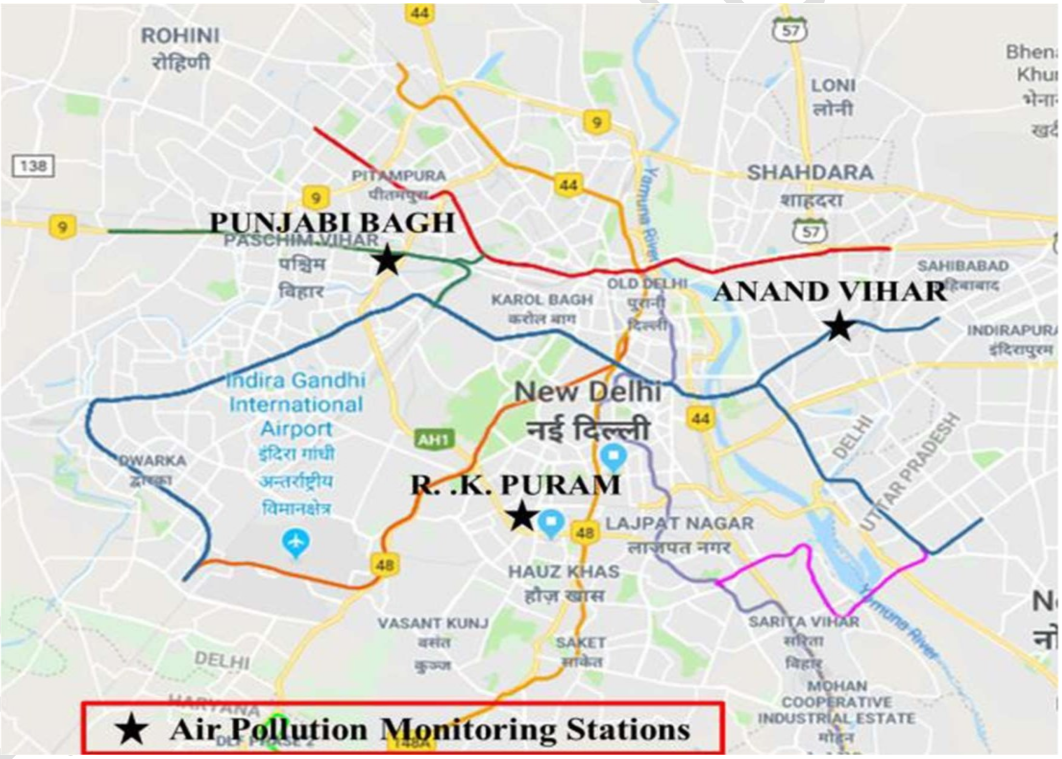


Figure 12: Simulated stations for air pollution monitoring from the dataset

VIII. Conclusion

This research study utilized a dataset comprising both numerical and visual data. The STLSTM AutoEncoder handled the numerical time series data, while the Concatenated Visual Geometry Group Neural Network (CVGGNet) models (CVGG16 and CVGG19) used image data to compare performance based on network depth. The results of the final investigation demonstrated that the deeper CVGG19 model performed up to 14.2% better than modality models with single data input that used only numerical data. The multimodal deep learning model, which integrated both types of data, significantly outperformed the models with single data input. In principle, our method can be utilised in any application that involves the connection of sensor and picture data, and the utilisation of both of these in conjunction with one another may be beneficial to the learning process. To be more specific, the application of our methods in circumstances where there are classes that are imbalanced in multi-modal classifications is extremely favourable. This encompasses the utilisation of hyperspectral pictures alongside with sensor data for the purpose of monitoring and diagnosing dangerous circumstances in a variety of sectors, including agriculture, geology, and environmental sciences. Optimization strategy has been planned to enhance the current models performance as future enrichments.

REFERENCES

1. Disease Control and Prevention Agency. Available online: http://www.kdca.go.kr/contents.es?mid=a20304030300 (accessed on 19 August 2022).

2. Ko, K.K.; Shahzad, E.S.J. Big data merging and deep learning model optimization for improving weather information forecasting performance. Inst. Electron. Inf. Eng. 2021, 58, 39–46.
3. Athira, V.; Geetha, P.; Vinayakumar, R.; Soman, K. Deepairnet: Applying recurrent networks for air quality prediction. Procedia Comput. Sci. 2018, 132, 1394–1403.
4. Chau, P.N.; Zalakeviciute, R.; Thomas, I.; Rybarczyk, Y. Deep Learning Approach for Assessing Air Quality During COVID-19 Lockdown in Quito. Front. Big Data 2022, 5, 842455. [PubMed]
5. Salman, A.G.; Heryadi, Y.; Abdurahman, E.; Suparta, W. Single layer and multi-layer long short-term memory (lstm) model with intermediate variables for weather forecasting. Procedia Comput. Sci. 2018, 135, 89–98.
6. Bekkar, A.; Hssina, B.; Douzi, S.; Douzi, K. Air-pollution prediction in smart city, deep learning approach. J. Big Data 2021, 8, 161. [PubMed]
7. Xie, H.; Ji, L.; Wang, Q.; Jia, Z. Research of PM2.5 Prediction System Based on CNNs-GRU in Wuxi Urban Area. IOP Conf. Ser. Earth Environ. Sci. 2019, 300, 032073.
8. Kalajdjieski, J.; Zdravevski, E.; Corizzo, R.; Lameski, P.; Kalajdziski, S.; Pires, I.M.; Garcia, N.M.; Trajkovik, V. Air pollution prediction with multi-modal data and deep neural networks. Remote Sens. 2020, 12, 4142.
9. Ministry of Environment. Available online: http://www.me.g.,o.kr/home/web/board/read.do?pagerOffset=0&maxPageItems=10&maxIndexPages=10&searchKey=&searchValue=&menuId=286&orgCd=&boardId=1485080&boardMasterId=1&boardCategoryId= 39&decorator= (accessed on 19 August 2022).
10. National Meteorological Satellite Center. Available online: http://wiki.nmsc.kma.go.kr/doku.php?id=homepage:gk2a:aep (accessed on 19 August 2022).
11. Goki, S. Deep Learning from Scratch2: Recurrent neural networks and natural language processing that are implemented and learned directly with Python; Hanbit Media: Seoul, Korea, 2017; pp. 191–287.
12. Hinton, G.E.; Salakhutdinov, R.R. Reducing the Dimensionality of Data with Neural Networks. Science 2006, 313, 504–507. [PubMed]

13. Goki, S. Deep Learning from Scratch: Deep learning theory and implementation in Python; Hanbit Media: Seoul, Korea, 2017; pp. 107–259.

14. Karen, S.; Andrew, Z. Very deep convolutional networks for large-scale image recognition. In Proceedings of the 3rd International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015; pp. 1–14.

15. Jiquan, N.; Aditya, K.; Mingyu, K.; Juhan, N.; Honglak, L.; Andrew, N. Multimodal deep learning. In Proceedings of the 28th international Conference on Machine Learning, Washington, DC, USA, 28 June–2 July 2011; pp. 689–696.

16. Bae, K.I.; Lee, Y.-S.; Lim, C.-W. Multi-view learning review: Understanding methods and their application. Korean J. Appl. Stat. 2019, 32, 41–68.

17. Mateus, B.C.; Mendes, M.; Farinha, J.T.; Assis, R.; Cardoso, A.M. Comparing LSTM and GRU models to predict the condition of a pulp paper press. Energies 2021, 14, 6958.

18. Lee, M.H.; Moon, G.-M.; Hong, S.-H.; Kim, H.-D. Kubeflow-If You Are New to Machine Learning in Kubernetes; Digital Books: Seoul, Korea, 2020; pp. 170–189.

19. James, B.; Yoshua, B. Random search for hyper-parameter optimization. JMLR 2012, 13, 281–30

20. Tsai, C.-F., Li, M.-L., Lin, W.-C.: 'A class center based approach for missing value imputation'; Knowledge-Based Systems, 151 (2018), 124–135. [USEPA, 2020] USEPA: 'US-EPA'; (2020).

21. Kulurkar, P., kumar Dixit, C., Bharathi, V. C., Monikavishnuvarthini, A., Dhakne, A., & Preethi, P. (2023). AI based elderly fall prediction system using wearable sensors: A smart home-care technology with IOT. *Measurement: Sensors*, *25*, 100614.

22. Enders, C. K.: 'A primer on maximum likelihood algorithms available for use with missing data'; Structural Equation Modeling, 8, 1 (2001), 128–141.

23. Andiojaya, A., Demirhan, H.: 'A bagging algorithm for the imputation of missing values in time series'; Expert Systems with Applications, 129 (2019), 10– 26.

24. Asokan, R., & Preethi, P. (2021). Deep learning with conceptual view in meta data for content categorization. In Deep Learning Applications and Intelligent Decision Making in Engineering (pp. 176-191). IGI global.

25. Preethi, P., Saravanan, T., Mohanraj, R., & Gayathri, P. G. (2024). A real-time environmental air pollution predictor model using dense deep learning approach in IoT infrastructure. Global NEST Journal, , Vol 26, No 3, 05666.