Major Project
Project Report On

"Time Series With LSTM In Machine Learning"

***Submitted* By**
Mohd Sameer Salmani (2K19/SWE/07)

MASTER OF TECHNOLOGY
IN
SOFTWARE ENGINEERING

Under the guidance
Of
Dr. Abhilasha Sharma
Assistant Professor,DTU

Department of Computer Science
Delhi Technological University, Delhi

June-2021

DELHI TECHNOLOGICAL UNIVERSITY
SHAHBAD DAULATPUR, DELHI-110042

DECLARATION

I hereby declare that the work presented in this report entitled "Comparing the predictive performances of different classification algorithm and ensemble techniques", in fulfillment of the requirement for the award of the MASTER OF TECHNOLOGY degree in Software Engineering submitted in Computer Science Department at DELHI TECHNOLOGICAL UNIVERSITY, New Delhi, is an authentic record of my own work carried out during my degree under the guidance of Dr. Abhilasha Sharma.

The work reported in this has not been submitted by me for the award of any other degree or diploma.

Date: 9thJune2021

Place: Delhi

Mohd Sameer Salmani (2K19/SWE/07)

CERTIFICATE

This is to certify that Mohd Sameer Salmani (2K19/SWE/07) has completed the project titled "Comparing the predictive performances of different classification algorithm and ensemble techniques" under my supervision in partial fulfillment of the MASTER OF TECHNOLOGY degree in Software Engineering at DELHI TECHNOLOGICAL UNIVERSITY.

Dr. Abhilasha Sharma
(Supervisor)
(Assistant Professor), DTU

ABSTRACT

Air pollution has numerous implications for agriculture, the economy, traffic accidents and health. In recent years, the fast development of business technology has been in the middle of severe environmental pollution. long-faced with several environmental pollution issues, particulate (PM2.5) that is the subject of special attention & is wealthy in an exceedingly sizable amount of toxic and harmful substances. additionally, PM2.5 features a long continuance within the atmosphere and an extended transport distance, that the analysis of the distribution of PM2.5 is a vital drawback for predicting the standard of the air.
Therefore, this project proposes a way supported a univariate remembering network (LSTM) to investigate the spatiotemporal characteristics of the distribution of PM2.5 so as to predict the air quality in many cities. within the prediction, information records of real words were collected and analyzed, and 3 measures of exactness (i.e. mean absolute error (MAE), mean root error (RMSE) and root error (MSE)) were wont to measure the performance of the tactic projected during this project. . For the analysis of the projected technique, the performance of the projected technique is compared to alternative machine learning strategies. The results of the sensible experiments show that the projected MAE, RMSE and MSE strategies square measure inferior to alternative machine learning strategies.

ACKNOWLEDGEMENT

I am very thankful to **Dr. Abhilasha Sharma** (Assistant Professor, Department of Computer Science) and all the faculty members of the Department of Computer Science at DTU. They all provided us with immense support and guidance for the project.

I would also like to express my gratitude to the University for providing us with the laboratories, infrastructure, testing facilities, and environment which allowed us to work without any obstructions.

I would also like to appreciate the support provided to us by our lab assistants, seniors, and our peer group who aided us with all the knowledge they had regarding various topics.

Mohd Sameer Salmani
Roll No. 2K19/SWE/07

INDEX

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

RNN	Recurrent Neural Network
CNN	Convolutional Neural Network
LSTM	Long Short Term Memory Model
ARIMA	Autoregressive Integrated Moving Average
TSF	Time Series Forecasting
HMM	Hidden Markov Model
SOL	Structure Of LSTM
DP	Data Preparation
DV	Data Visualization
FG	Forged Gate
IP	Input Gate
OP	Output Gate
HMOCM	HeatMap Of Correlation Matrix
MP	Manipulated Data
HL	Hidden Layer
OD	Original Data
MSE	Mean Square Error
MAE	Mean Absolute Error
RMSE	Root Mean Square Error
MAPE	Mean Absolute Percentage Error

CHAPTER 1

INTRODUCTION

A set time may be a series of notes in an exceedingly specified fundamental quantity. A univariate time set includes values loving unmatched variables in an exceedingly cyclic time state in an exceedingly single fundamental quantity, and a variable time set includes values loving over one variable within the same cyclic time state in an exceedingly single fundamental quantity. the best example of a statistic that everybody finds may be a amendment in temperature earlier or later than on a daily basis, week, month, or year.

Data Processing And Visualization

A statistic may be a sequence of observations obtained at a particular time. Therefore, order and continuity should be maintained in any program. knowledge | the info | the information} we'll use may be a statistic network with data of around one hour, on air quality in contaminated Italian cities. it's necessary to confirm that:

1)The statistic is equally divided

2)Absence of unwanted worth or area. If the statistic isn't continuous, we will take a sample or cut back the sample as follows:-

Showing df.head()

import pandas

df = pandas.read_csv("AirQualityUCI.csv", sep = ";", decimal = ",")

df = df.iloc[: , 0:14]

len(df)

```
9471
```

df.head()

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T	RH
0	10/03/2004	18.00.00	2.6	1360.0	150.0	11.9	1046.0	166.0	1056.0	113.0	1692.0	1268.0	13.6	48.9
1	10/03/2004	19.00.00	2.0	1292.0	112.0	9.4	955.0	103.0	1174.0	92.0	1559.0	972.0	13.3	47.7
2	10/03/2004	20.00.00	2.2	1402.0	88.0	9.0	939.0	131.0	1140.0	114.0	1555.0	1074.0	11.9	54.0
3	10/03/2004	21.00.00	2.2	1376.0	80.0	9.2	948.0	172.0	1092.0	122.0	1584.0	1203.0	11.0	60.0
4	10/03/2004	22.00.00	1.6	1272.0	51.0	6.5	836.0	131.0	1205.0	116.0	1490.0	1110.0	11.2	59.6

To method the statistic, we tend to make sure that there's no NaN (NULL) worth within the information set; If it exists, we will replace it with a worth of zero, average, previous or posterior. Replacement is healthier than deletion to take care of the continuity of the statistic. However, in our information set, these last values seem to be zero and thus their decrease won't have an effect on continuity.

Dropping NaN(Not-a-Number)

df.isna().sum()

```
Date              114
Time              114
CO(GT)            114
PT08.S1(CO)       114
NMHC(GT)          114
C6H6(GT)          114
PT08.S2(NMHC)     114
NOx(GT)           114
PT08.S3(NOx)      114
NO2(GT)           114
PT08.S4(NO2)      114
PT08.S5(O3)       114
T                 114
RH                114
dtype: int64
```

df = df[df['Date'].notnull()]

df.isna().sum()

```
Date              0
Time              0
CO(GT)            0
PT08.S1(CO)       0
NMHC(GT)          0
C6H6(GT)          0
PT08.S2(NMHC)     0
NOx(GT)           0
PT08.S3(NOx)      0
NO2(GT)           0
PT08.S4(NO2)      0
PT08.S5(O3)       0
T                 0
RH                0
dtype: int64
```

Time series ar typically premeditated as a line graph against the clock. So, currently we're getting to merge the datetime columns and convert them to datetime objects from the string. this may be done victimisation the date and time library.

Converting to datetime object

df['DateTime'] = (df.Date) + ' ' + (df.Time)

print (type(df.DateTime[0]))

<class 'str'>

import datetime

df.DateTime = df.DateTime.apply(lambda x: datetime.datetime.strptime(x, '%d/%m/%Y %H.%M.%S'))

print (type(df.DateTime[0]))

<class 'pandas._libs.tslibs.timestamps.Timestamp'>

Let’s see how some variables like temperature change over time..

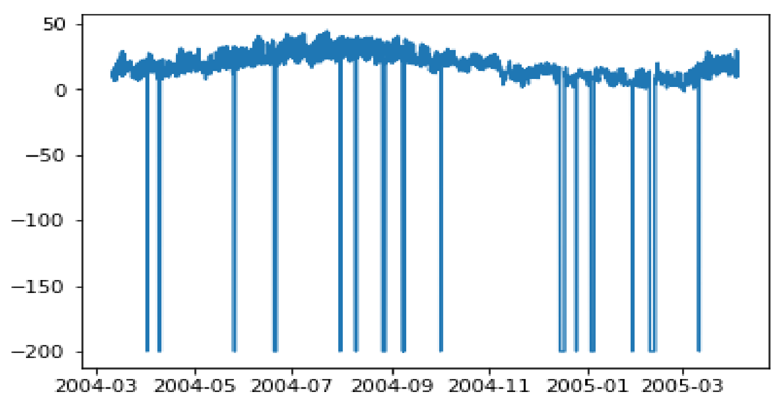
Showing plots

df.index = df.DateTime

import matplotlib.pyplot as plt

plt.plot(df['C6H6(GT)'])

```
[<matplotlib.lines.Line2D at 0x1eaad67f780>]
```

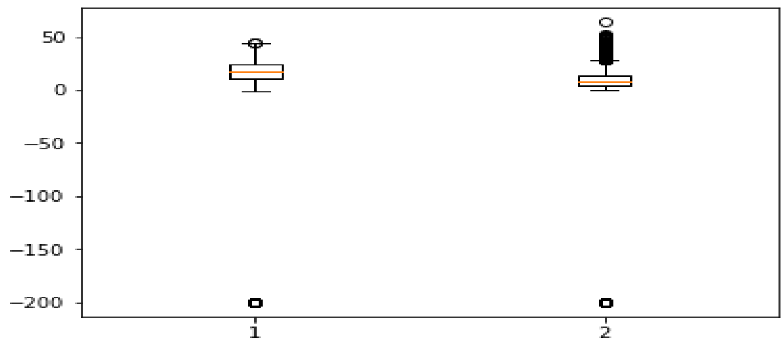


Box charts ar another helpful form of chart that permit you to collect a great deal of data a couple of cluster of information into one chart. Displays mean, twenty fifth and seventy fifth quartiles, and descriptions for one or a lot of variables. If the quantity of outliers is little and much from average, we will take away the outliers by setting them to a mean or mark of seventy fifth.

Showing Boxplots

plt.boxplot(df[['T','C6H6(GT)']].values)

```
{'whiskers': [<matplotlib.lines.Line2D at 0x1eaac16de80>,
   <matplotlib.lines.Line2D at 0x1eaac16d908>,
   <matplotlib.lines.Line2D at 0x1eaac177a58>,
   <matplotlib.lines.Line2D at 0x1eaac177cf8>],
   'caps': [<matplotlib.lines.Line2D at 0x1eaac16d2b0>,
   <matplotlib.lines.Line2D at 0x1eaac16d588>,
   <matplotlib.lines.Line2D at 0x1eaac1a69e8>,
   <matplotlib.lines.Line2D at 0x1eaac1a64a8>],
   'boxes': [<matplotlib.lines.Line2D at 0x1eaac16dc50>,
   <matplotlib.lines.Line2D at 0x1eaac1779b0>],
   'medians': [<matplotlib.lines.Line2D at 0x1eaac16d4a8>,
   <matplotlib.lines.Line2D at 0x1eaac1a6c50>],
   'fliers': [<matplotlib.lines.Line2D at 0x1eaac177dd8>,
   <matplotlib.lines.Line2D at 0x1eaac1a6c18>],'means': []
}
```



The written account series has four elements as indicated below -

1. Level - this is often the typical worth at that the series varies.
2. Tendency - this is often a rise or decrease within the behavior of a variable over time.
3. Season: this is often a written account series cycle behavior.
4. Noise - this is often a mistake in extra observations because of environmental factors.

Time series modeling techniques

To capture this element, there ar variety of widespread statistic modeling techniques. This section provides a quick introduction to every technique, however we'll discuss them very well within the next chapter –

Naive method

This is an easy estimation technique as a result of the expected worth receives {a worth | a worth | a price} adequate to the previous value of a time variable quantity, or the previous actual worth. this is often used for comparison with refined modeling techniques.

Automatic regression

Automatic regression predicts the worth of a future term supported the worth of the previous term. Automatic regression predictions might match the information higher than naive ways, however might not be able to make a case for the season.

The ARIMA model

The integrated regressive automatic moving average models {the worth | the worth} of the variable as a linear operate of the previous value and also the residual error on previous temporal measurements of stationary time. However, universe information will be stationary and seasonal, thus Seasonal-ARIMA and Fractional- ARIMA are developed. ARIMA works on a univariate statistic, to method the various variables VARIMA is introduced.

Exponential launch

It models {the worth | the worth} of the variable as associate degree exponentially weighted linear operate of the previous value. This applied mathematics model may handle trends and seasonality.

LSTM

The immediate memory Model (LSTM) may be a repetitive neural network used for statistic to clarify long-run addictions. It will be trained with massive amounts of information to capture trends in varied statistic.

CHAPTER 2

THEORY

The LSTM network is associate extension of the perennial Neural Network (RNN) introduced primarily to influence things wherever RNNs fail. Speaking of RNN, this can be a network that uses the present input by considering the previous output (feedback) and storing it in its memory for a brief time (short term memory). Among its numerous applications, the foremost widespread square measure within the fields of speech process, non- stochastic process management, and music composition. Even so, there square measure downsides to RNN. First, it fails to store info for extended periods of your time. generally a regard to specific info keep for a protracted time is required to predict the present output. however the RNN is by no means that able to address such a “long-term dependence”. Second, there's no higher management over that elements of the context should continue and that a part of the past should be "forgotten". Another downside with RNNs is that the explosion and disappearance of gradients (explained later) that occur throughout the method of network formation through rollback. Thus, short term memory (LSTM) is taken under consideration. it absolutely was designed in such some way that the gradient downside nearly utterly disappears, whereas the coaching pattern remains unchanged. The long delays of some issues square measure blocked exploitation LSTM wherever they additionally influence noise, distributed illustration and constant values. With LSTM, it's not necessary to store a finite variety of conditions from the previous one pro re nata within the Hidden mathematician Model (HMM). LSTM provides USA numerous parameters like learning rate and entry and exit bias. Therefore, no refined adjustment is critical. The complexness of change every weight is reduced to O (1) with LSTM, like Back Propagation Through Time (BPTT), that may be a profit.
LSTM was unconcealed by Hochreiter & Schmidhuber. It overcomes the matter of long dependency of RNNs wherever RNNs willnot predict words keep in LTM however can give additional correct predictions from up-to-date info. thanks to the augmented spacing distance, the RNN doesn't give economical performance. LSTM by default will store this info for a protracted time. it's wont to method, predict and rank supported statistic information.

Gradients that explode and disappear:
During the network learning method, the most objective is to reduce the losses (in terms of errors or costs) discovered within the output once the coaching information is distributed thereto. we have a tendency to calculate the gradient, that is, the loss with regard to a selected weight cluster, change the we have a tendency toights consequently associated repeat this method till we get an optimum set of weights with marginal losses. this can be the conception of going back. generally the gradient is sort of negligible. It ought to be noted that the gradient of the layers depends on sure parts of the serial layers. If a number of these parts square measure little (less than 1), the result obtained, i.e. the gradient, are going to be smaller. this can be referred to as the dimensions result. once this gradient is increased by the educational rate that itself may be a little worth between zero.1 and 0.001, it produces a smaller worth. As a result, the modification in weight is very little, manufacturing nearly identical output as before. Likewise, if the gradient is sufficiently massive in worth thanks to the massive worth of the part, the load is reduced to {a worth|a worth|a price} prodigious the optimum value. this can be referred to as the explosive gradient downside. To avoid this scaling result, the neural network unit is reconstructed such the scaling issue remains one. The cell is then enriched with many gate units and is termed LSTM.

Structure Of LSTM :- The LSTM incorporates a chain structure that contains four totally different neural networks and blocks of memory referred to as cells.

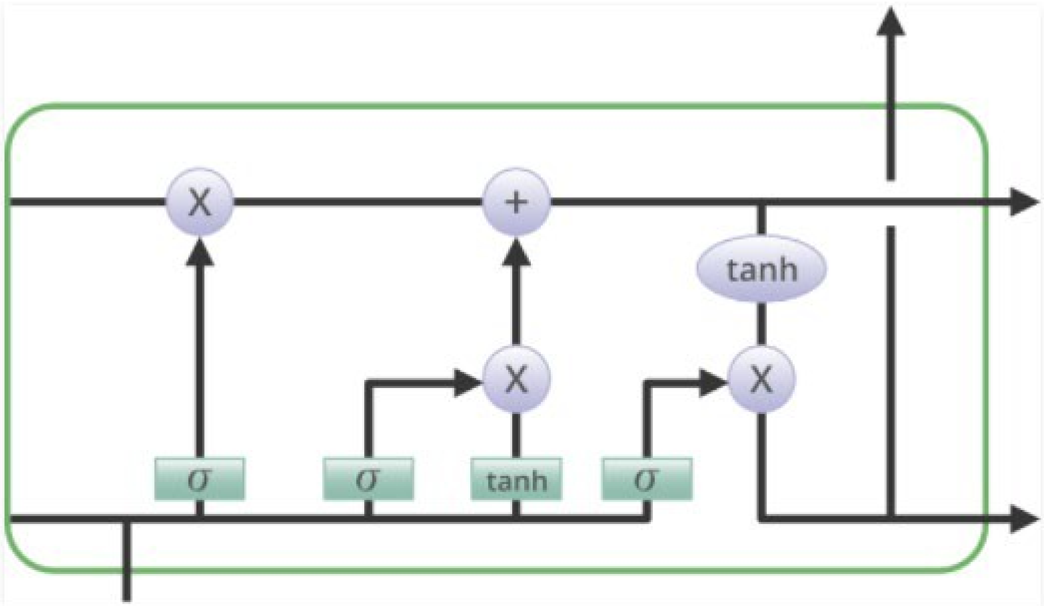


Fig.1 Structure Of LSTM

Information is keep by cells and memory manipulation is performed by gateways. There square measure 3 doors –

1) Forget Gate: info that's now not helpful within the state of the cell is deleted with a forgetting gate. 2 inputs x_t (input at a given time) and h_t-1 (previous cell output) square measure sent to the gate and increased by the load matrix followed by the addition of the bias. The result goes through associate activation operate that provides a binary output. If for a selected cell state the output is zero, info is forgotten and for output one, this info is keep for future use.

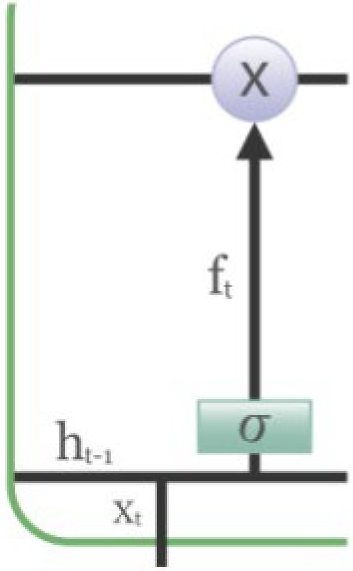


Fig.2 Forged Gate

2) Input Gate: The addition of helpful info to the state of the cell is finished with associate entrance door. First, the data is organized exploitation the sigmoid operate and filters the values to be remembered within the same approach because the forgetting gates exploitation the h_t-1 and x_t inputs. Then a vector is formed exploitation the tanh operate which provides associate output of -1 to +1, containing all potential values of heated up -1 and x_t. Atlast, vector values and controlled values square measure increased to urge helpful info

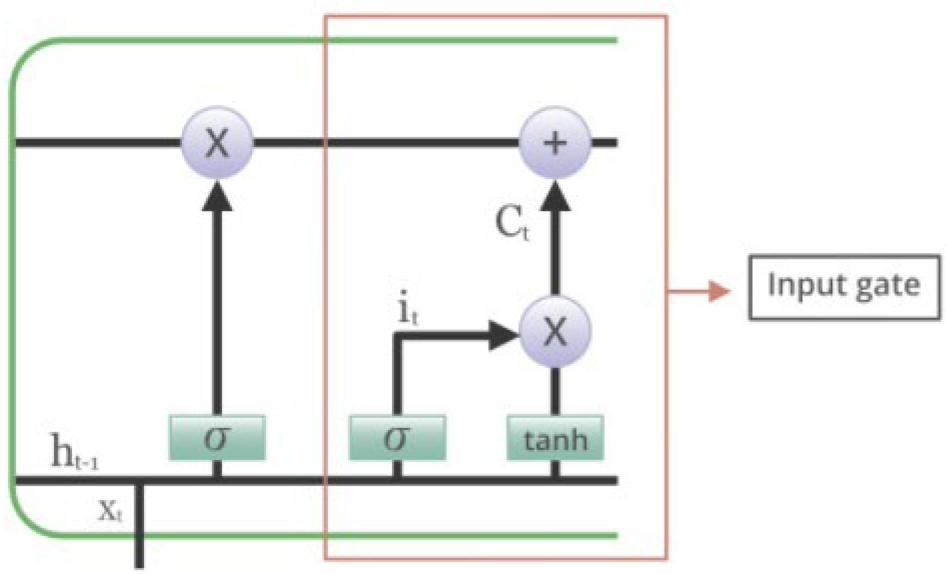


Fig.2 Input Gate

3) Output Gate: The task of extracting helpful info from the present state of the cell to be output is performed by the door output. First, a vector is generated by applying the tanh operate to the cell. Then the data is organized exploitation the sigmoid operate and filters the values to be remembered exploitation the h_t-1 and x_t inputs. Atlast, vector values and controlled values square measure increased to be output and input to following cell.

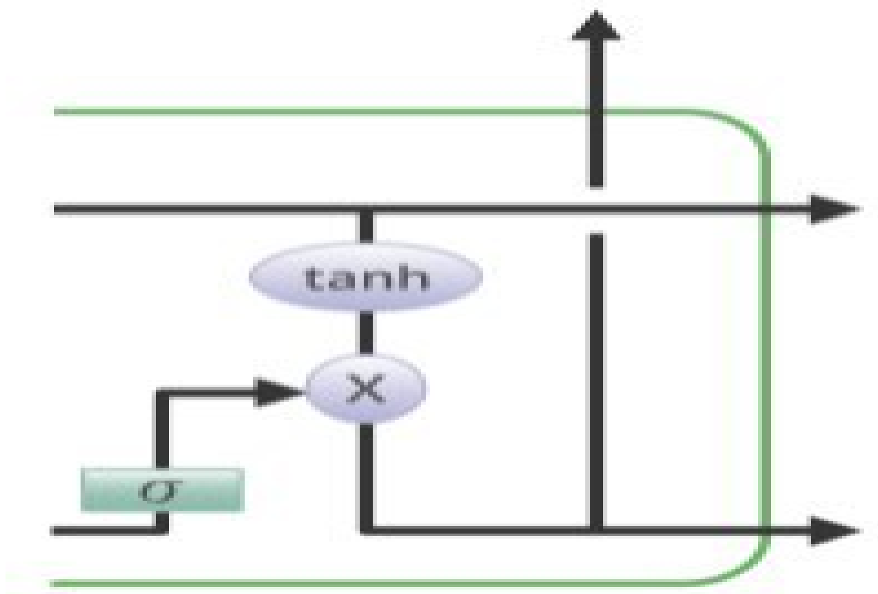


Fig.4 Output Gate

Architecture

The fundamental distinction between RNN and LSTM architectures is that the hidden layer of LSTM may be a closed unit or cell. These square measure four layers that move with one another to supply the output of the cell yet because the state of the cell. These 2 parts square measure then passed to following hidden layer. not like RNNs that acquire one somatic cell tanh layer, LSTM consists of 3 sigmoid logistical gateways and a tanh layer. Gates was introduced to limit the data carried through cells. They confirm what half of} the data are going to be needed by following cell and what part are going to be deleted. The output is sometimes between zero and one wherever "0" means that "minus all" and "1" means that "include all".

Hidden Layers Of LSTM :

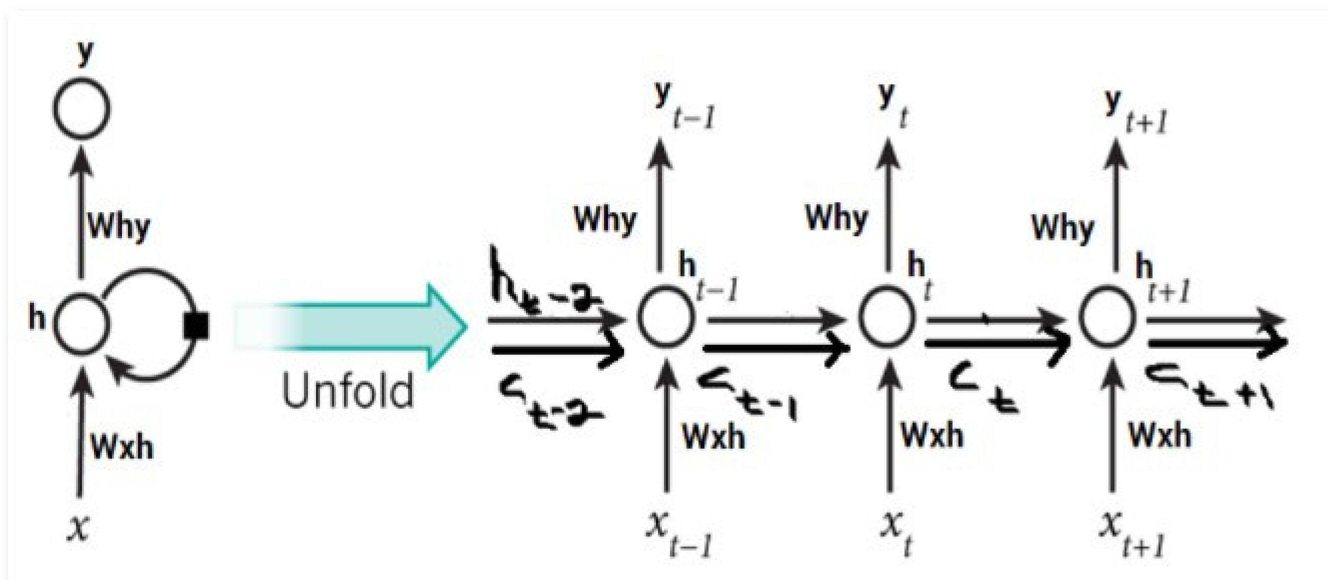


Fig.5 Hidden Layer Of LSTM

Each LSTM cell has 3 inputs angular distance , metallic element and x_t and 2 outputs heated up and cathode-ray tube. For a given time t, heated up is that the hidden state, cathode-ray tube is that the state of the cell or of the memory, x_t is that the current information or the entry purpose. the primary sigmoid layer has 2 inputs-h_ and x_t wherever angular distance is that the hidden state of the previous cell. it's called the forgetting gate as a result of its output selects the number of previous cell info to incorporate. The output may be a variety in [0,1] increased (point by point) by the previous cell state metallic element .

Conventional LSTM :

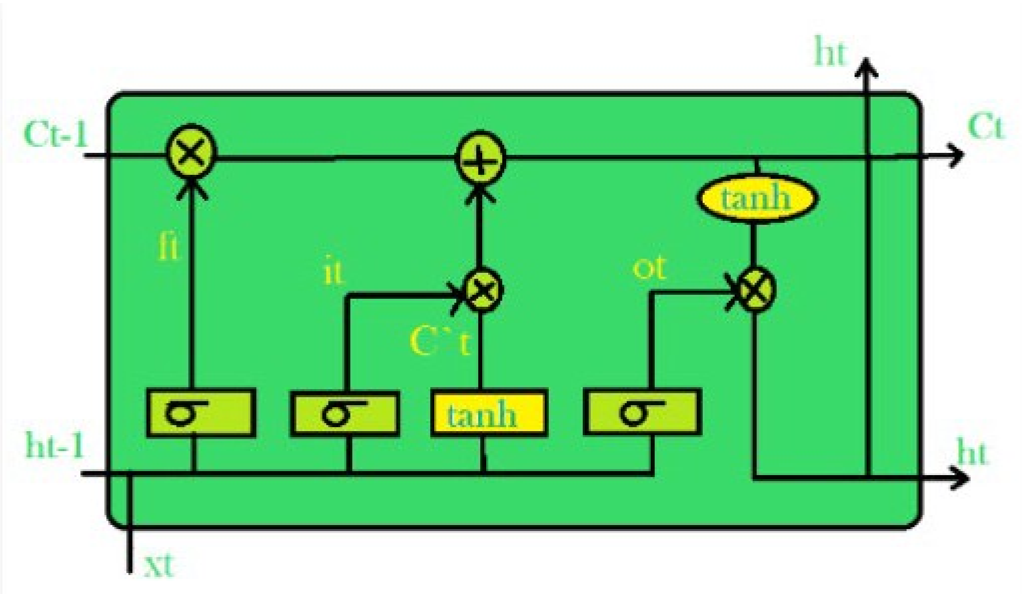


Fig.6 Conventional LSTM

The second sigmoid layer is that the entranceway that determines what new info are going to be more to the cell. It needs 2 inputs angular distance and x_t. The tanh layer creates a vector cathode-ray tube of the new candidate worth. Together, these 2 layers confirm what info to store within the cell state. The multiplication of their points (i_t? C_t) tells USA the number of data which will be more to the state of the cell. The result's then more by the results of the forgetting gate increased by the previous cell state (f_t * metallic element) to supply the present cathode-ray tube cell state. Then, cell output was calculated exploitation the sigmoid and tanh layers. The sigmoid layer decides what a part of the cell state are going to be gift within the output whereas the tanh layer transfers the output to the vary [-1.1]. The results of each layers bear purpose multiplication to supply the output of the ht cell.

Applications:

The LSTM model ought to be trained with a coaching dataset before it will be utilized in real applications. a number of the additional stringent applications square measure represented below:

1. Language modeling or text production, that involves word count once word sequences square measure input. Language models will be exploited at the character, n-gram, sentence or perhaps paragraph level.
2. Image process, that consists of activity a picture analysis and closing the leads to sentences. For this, it's necessary to possess an information set composed of an oversized variety of pictures with acceptable descriptive descriptions. The trained model was wont to predict the characteristics of the pictures found within the dataset. this can be photographic information. The dataset is then processed in order that solely the foremost impactful words square measure there. this can be matter information. exploitation these 2 styles of information, we have a tendency to tried to suit the model. The results of the model's work is to supply a descriptive sentence for a picture one word at a time taking the input word antecedently expected by the model yet because the image recognition of speech and handwriting.

3. Music generation that is incredibly like text generation wherever LSTM predicts musical notes rather than text by analyzing the mix of notes given as input.
4. Linguistic translation involves the mapping of sequences in one language with sequences in another language. a bit like image process, a dataset, containing sentences and their translations, is 1st cleansed up and solely a vicinity is employed to coach the model. The LSTM encoder model is employed, that 1st converts the input sequence to its vector illustration (encoding) so outputs it in its translated version.

Disadvantages:

As said, everything during this world has its own execs and cons, LSTM additionally incorporates a few cons that square measure mentioned below:

1. LSTM became widespread as a result of it might solve the matter of missing gradients. however it clad that they did not take away it utterly. the matter is that the info still must be transferred from cell to cell for analysis. additionally, cells became terribly complicated currently with extra options (such as doors) more to the image.
2. they have a great deal of resources and time to be trained and ready for globe applications. Technically, they need high information measure memory as a result of linear layers square measure gift in each cell and don't seem to be ordinarily provided by the system. So, in terms of hardware, LSTM becomes inefficient.
3. With the rise in data processing, developers square measure trying to find models which will bear in mind past info longer than LSTM. The inspiration for such a model is that the human habit of breaking sure info into little elements in order that they're straightforward to recollect.
4. LSTMs square measure plagued by totally different random weight initializations and thus behave within the same approach as feedforward neural networks. they like little starts.
5. LSTM tends to be redundant and it's tough to use off-hook algorithms to limit this downside. Dropout may be a methodology of regulation within which perennial inputs and connections to the LSTM unit square measure possible excluded from activating and gaining weight throughout network coaching.

RELATED WORK

Air quality observance has been a motivating topic and plenty of studies are administered to use the IoT approach for air quality observance.
[2], [3], [4] They planned associate degree IoT platform to beat low exactness and low sensitivity.

[5] they need developed automaton apps to teach individuals concerning air quality. IoT isn't concerning the flow of information from sensors. There area unit several area unitas that are closely associated with IoT, like traffic management, healthcare, and others.

Several studies are administered to predict air observance knowledge to work out trends in pollution concentrations. [1] Recommendations for the prediction of pollution with neuro-fuzzy infrared systems. It combines each neural networks and symbolic logic principles.

Several studies have planned the ARIMA model to predict energy demand, temperature and epidemic diseases. [7] [8] [9]

Wang et al. [10] sculpturesque statistic knowledge with seasonal ARIMA to investigate pollution off the coast of the u. s..

The hunt for prognosticative statistic knowledge has additionally been developed exploitation neural network models.
In [11], ancient machine learning and neural networks mix to predict air quality
[12] proposed a technique to predict PM2.5 contamination. Wang et al.
[13] have planned the LSTM neural network to predict water quality. Alhirmizy et al.
[14] used the LSTM model for variable statistic prognostication for pollution in national capital, Spain.
This project proposes to develop associate degree LSTM model to predict air quality for univariate statistic knowledge.

CHAPTER 3

PROPOSED WORK

Steps Followed :-

- 1.Data preparation
- 2.Data visualization
- 3.LSTM data preparation
- 4.Model according to the regularization term
- 5.Evaluate model

Data Preparation

1. Replace the NA value
2. Analyze the date and time in a panda data frame index
3. An explicit name specified for each column

Data Visualization

1.Used matplotlib to plot

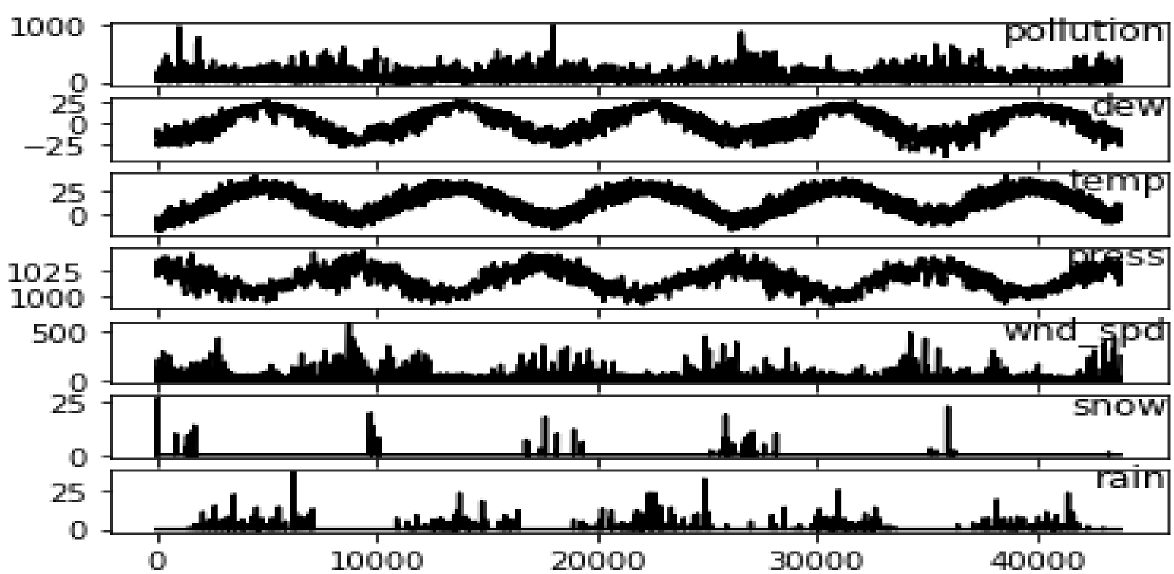


Fig.7 Data Visualization

LSTM Data Preparation

1.Data is normalized

2.Turning a dataset into a supervised learning problem

Original Data.

year_month_day_hour	pm2.5	DEWP	TEMP	PRES	cbwd	Iws	Is	Ir
2010-01-01 00:00:00	NaN	-21	-11.0	1021.0	NW	1.79	0	0
2010-01-01 01:00:00	NaN	-21	-12.0	1020.0	NW	4.92	0	0
2010-01-01 02:00:00	NaN	-21	-11.0	1019.0	NW	6.71	0	0
2010-01-01 03:00:00	NaN	-21	-14.0	1019.0	NW	9.84	0	0
2010-01-01 04:00:00	NaN	-20	-12.0	1018.0	NW	12.97	0	0

Fig.8 Original Dataset

Manipulated Data.

date	pollution	dew	temp	press	wnd_dir	wnd_spd	snow	rain
2010-01-02 00:00:00	129.0	-16	-4.0	1020.0	SE	1.79	0	0
2010-01-02 01:00:00	148.0	-15	-4.0	1020.0	SE	2.68	0	0
2010-01-02 02:00:00	159.0	-11	-5.0	1021.0	SE	3.57	0	0
2010-01-02 03:00:00	181.0	-7	-5.0	1022.0	SE	5.36	1	0
2010-01-02 04:00:00	138.0	-7	-5.0	1022.0	SE	6.25	2	0

Fig.9 Manipulated Dataset

HeatMap Of Correlation Matrix.

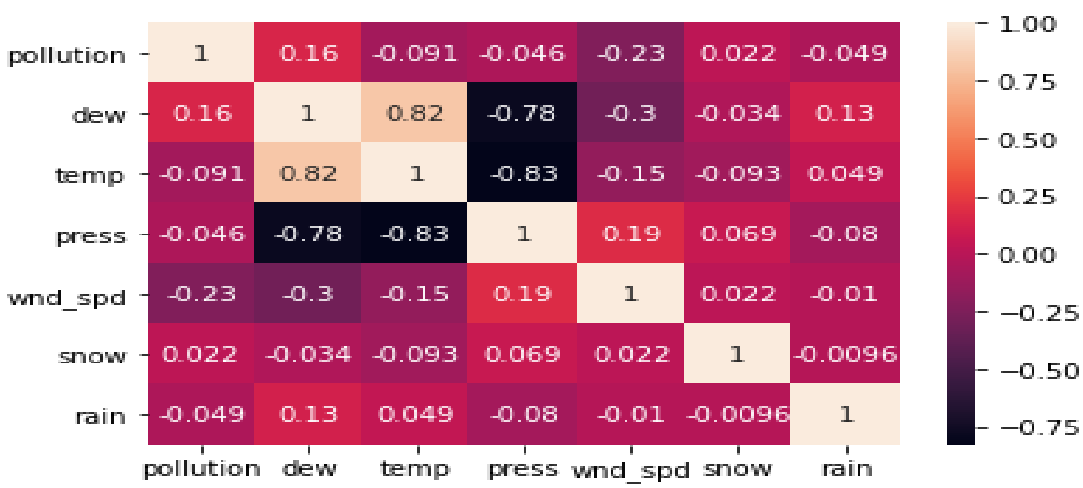


Fig.10 HeatMap Of Correlation Matrix

What is LSTM?
LSTM may be a category of repeated neural networks. Therefore, before you'll navigate the LSTM, it's necessary to know neural networks and repeated neural networks.

Neural networks
Artificial neural networks area unit stratified structures of connected neurons, galvanized by biological neural networks. Not associate degree rule however a mix of varied algorithms that permit US to perform advanced operations on knowledge.

Recurrent neural networks
It is a category of neural networks tailored to manage temporal knowledge. RNN neurons have a cell / memory state, and therefore the input is processed supported this internal state, that is obtained employing a loop within the neural network. There area unit continuance tanh layer modules in RNN that permit them to store data. However, shortly this can be why we want the LSTM model.

LSTM
They are special RNN capable of learning semipermanent knowledge dependencies. this can be achieved as a result of the continuance modules of the model have a mix of 4 layers that act with one another. Short- term and remembering may be a form of repetitive neural network. In the RNN, the output of the last step is entered because the input to this step.

Univariate LSTM model - LSTM may be accustomed model univariate statistic prediction issues.
we have a tendency to used some variations of the LSTM model for predicting univariate statistic.
This section is split into six sections :

1. Data Preparation
2. Vanilla LSTM
3. Stacked LSTM
4.Bidirectional LSTM
5.CNN LSTM
6. ConvLSTM

1.Data Preparation

Before a univariate series may be sculpturesque, it should be completed. The LSTM model can study a perform that maps the sequence of passed observations as input to the output observation. Therefore, the sequence of observations ought to be transformed into many examples which will be investigated by the LSTM.

2. Vanilla LSTM

Vanilla LSTM is associate degree LSTM model that features a single layer of hidden LSTM units associate degreed an output layer accustomed build predictions.

3. Stacked LSTM

The different hidden LSTM layers may be stacked on high of every different in what's referred to as the stacked LSTM model.

The LSTM layer needs three-dimensional input and therefore the default LSTM can turn out two-dimensional output as associate degree interpretation of the tip of the sequence. we will overcome this by distribution associate degree LSTM output worth to every time step within the computer file by setting return_sequences = true argument on the layer. this permits US to own a 3D output of the LSTM layer hidden at the input of successive one.

4. Bidirectional LSTM

On some sequence prediction issues, it should be advantageous for the LSTM model to check the forward and backward input sequences and mix the 2 interpretations. this can be referred to as two-way LSTM.

We can implement two-way LSTM for univariate statistic prediction by wrapping the primary hidden layer in associate degree envelope layer referred to as two-way.

5. CNN LSTM

Convolutional Neural Networks, or CNN for brief, area unit a sort of neural network developed to figure with two-dimensional image knowledge. CNN may be terribly economical at extracting and mechanically learning options from one-dimensional sequence knowledge like univariate statistic knowledge.

The CNN model may be utilized in a hybrid model with associate degree LSTM backend wherever CNN is employed to interpret the input sequences that area unit provided along because the LSTM model sequence to be taken. This hybrid model is termed CNN-LSTM.

The first step is to separate the input sequences into sequences which will be processed by the CNN model. for instance, we will 1st divide our univariate statistic knowledge into input / output samples with four input steps and one output. every sample will then be divided into 2 sub-samples, every with 2 time steps. CNN is in a position to interpret every 2 time step sequence and assign a statistic of event interpretations to the LSTM model to be processed as input. We can produce this parameter and specify range | the amount | the quantity} of ends as n_seq and therefore the number of your time steps per following as n_steps.

The computer file will then be reconstructed to own the specified structure :

[samples, subsequences, timesteps, features]

6. ConVLSTM

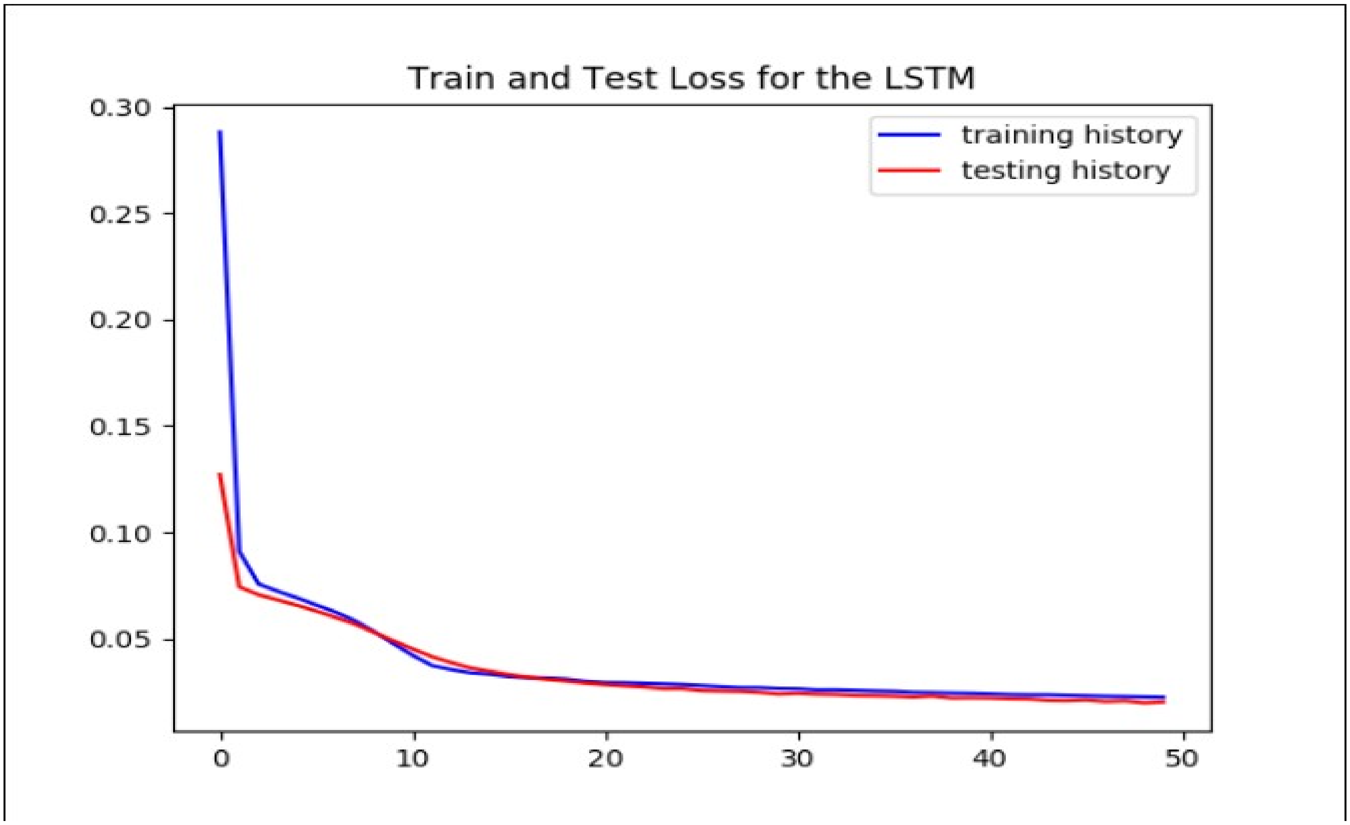
The type of LSTM related to CNN-LSTM is ConvLSTM, during which a standard input reading is constructed directly into every LSTM unit. ConvLSTM was developed to scan two-dimensional spatiotemporal knowledge, however may be tailored to be used with univariate statistic predictions.

The layer expects the input to be a sequence of two-dimensional pictures, that the form of the computer file ought to be:

[samples, timesteps, rows, columns, features]

Model according to the regularization term

1. Enter the data in the train and test
2. Divided into i / p and o / p
3. Reshape in 3D
4.Determine 50 neurons followed by 1 LSTM neuron
5. Increase dropout by 30%
6. Trace the history of training and test losses



Evaluate Model

It is necessary for US to live the performance of the model to use as feedback and comparison. totally different error measures area unit out there.

1. Mean Square Error
2. Root Mean Square Error
3. Mean Absolute Error
4. Mean Percentage Error
5. Mean Absolute Percentage Error

Mean Square Error

This is the square average of the distinction between the anticipated worth and therefore the actual worth. Sklearn provides this as a perform. it's a similar unit of true worth and sq. prediction and is often positive.

$$\text{MSE} = \frac{1}{n} \sum_{t=1}^{n} (y' - y^{t})^2$$

Where, y'_t is the predicted value

y^t is the actual value, and

n is the total number of values in test set.

It is clear from the equation that MSE is a lot of penal for offenses or a lot of necessary determinants.

Root Mean Square Error

This is the root of the equal error. it's additionally invariably positive and is within the knowledge vary.

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{t=1}^{n} (y'_t - y^{t})^2}$$

Where, y'_t is the predicted value

y^t is the actual value

n is total number of values in test set

It is the strength of unity and thus a lot of explicable compared to the MSE. The RMSE is additionally a lot of penal for larger infractions. we have a tendency to used RMSE metrics in our tutorial.

Mean Absolute Error

This is the common of absolutely the distinction between the anticipated worth and therefore the actual worth. it's a similar unit because the foretold and true worth and is often positive.

$$\text{MAE} = \frac{1}{n} \sum_{t=1}^{t=n} |y'_t - y_t|$$

Where y'_t is the predicted value

y_t is the actual value

n is total number of values in test set

Mean Absolute Percentage Error

This is the share of the common absolute distinction between the anticipated worth and therefore the actual worth, divided by the particular worth.

$$\text{MAPE} = \frac{1}{n} \sum_{t=1}^{n} \frac{y'_t - y_t}{y_t} * 100\%$$

Where y'_t is the predicted value

y_t is the actual value

n is total number of values in test set

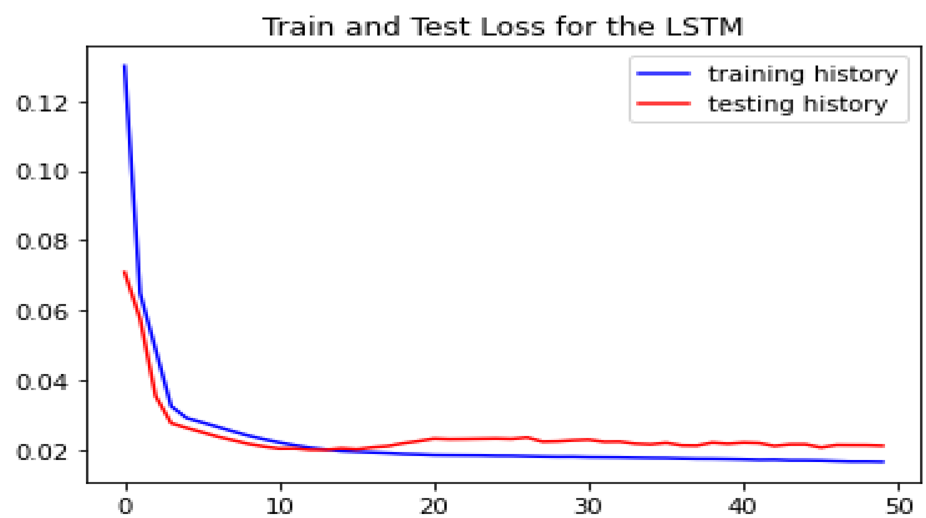
However, the draw back of exploitation these errors is that positive errors and negative errors will balance one another out. therefore the common absolute error share used.

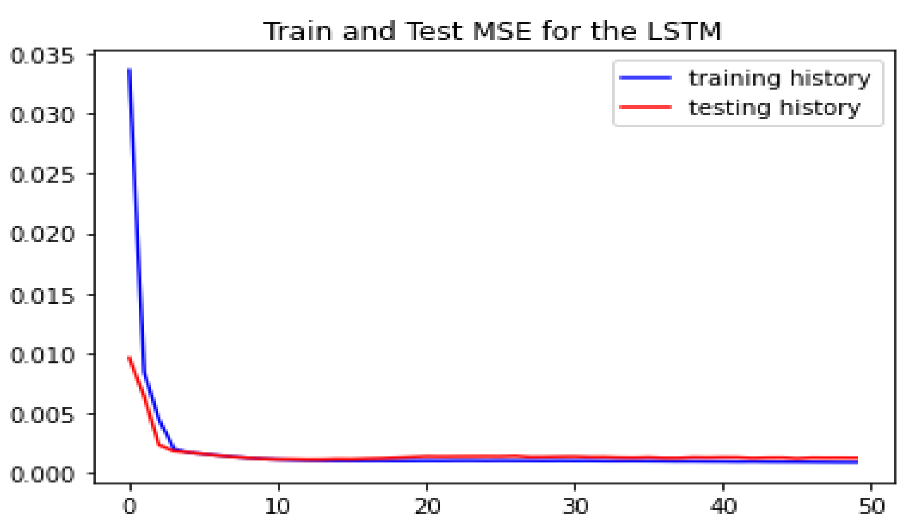
CHAPTER 4

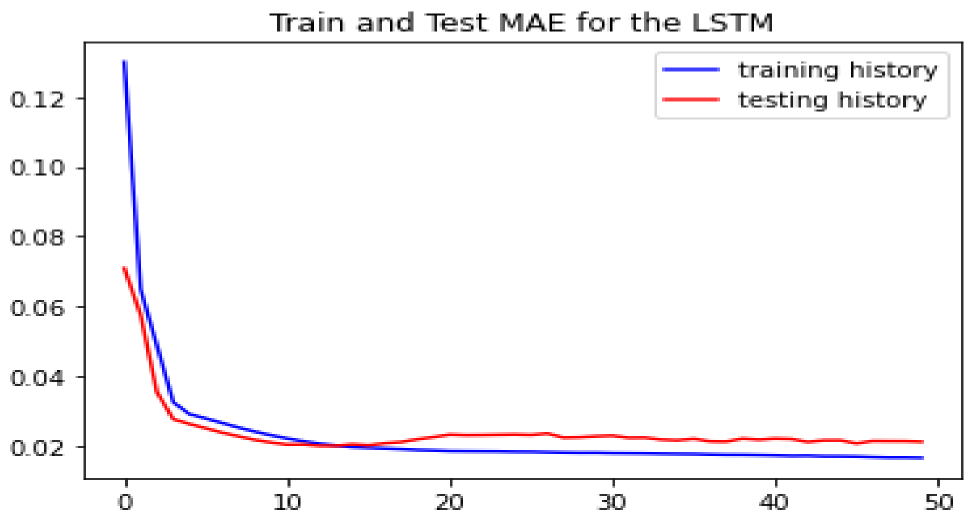
RESULTS

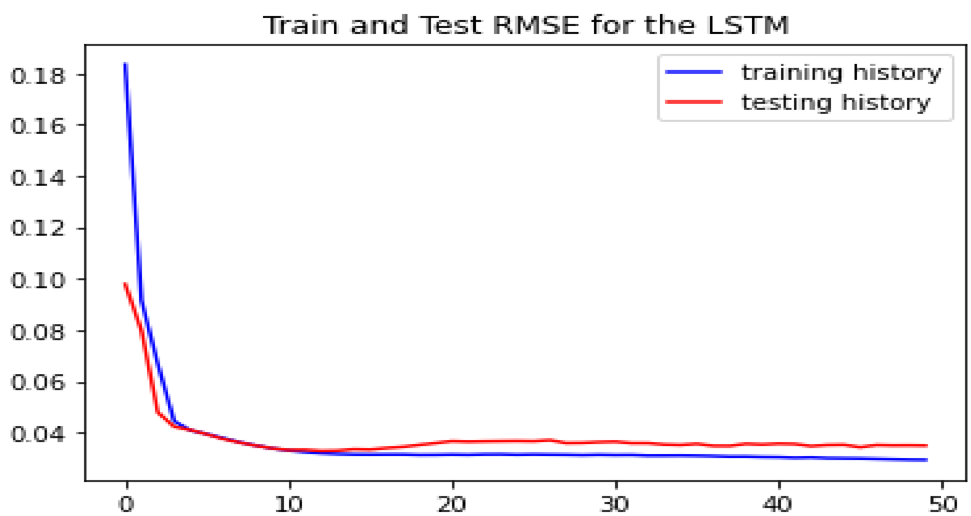
In this Chapter, we will see the results of the different Univariate LSTM Models. We will see the results of different LSTM Models for Time Series Forecasting.

1. Vanilla LSTM



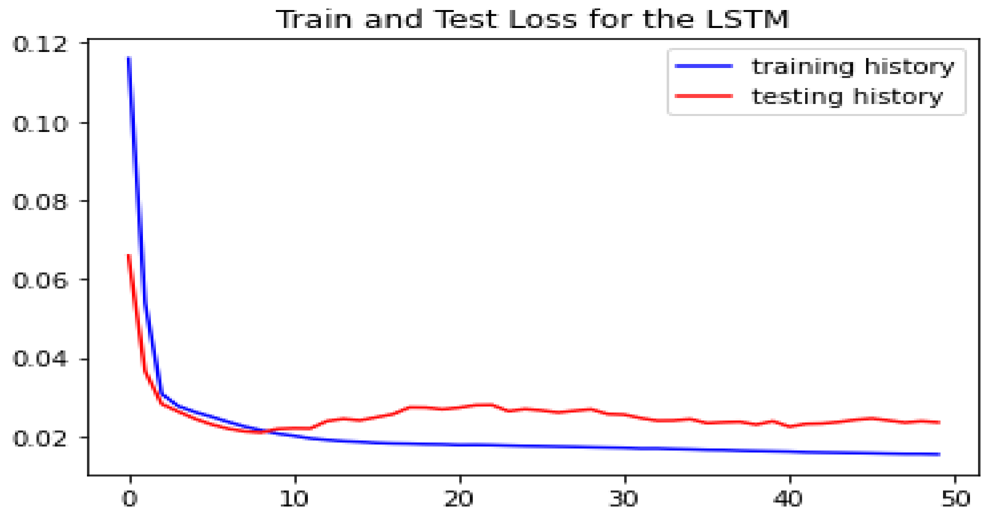


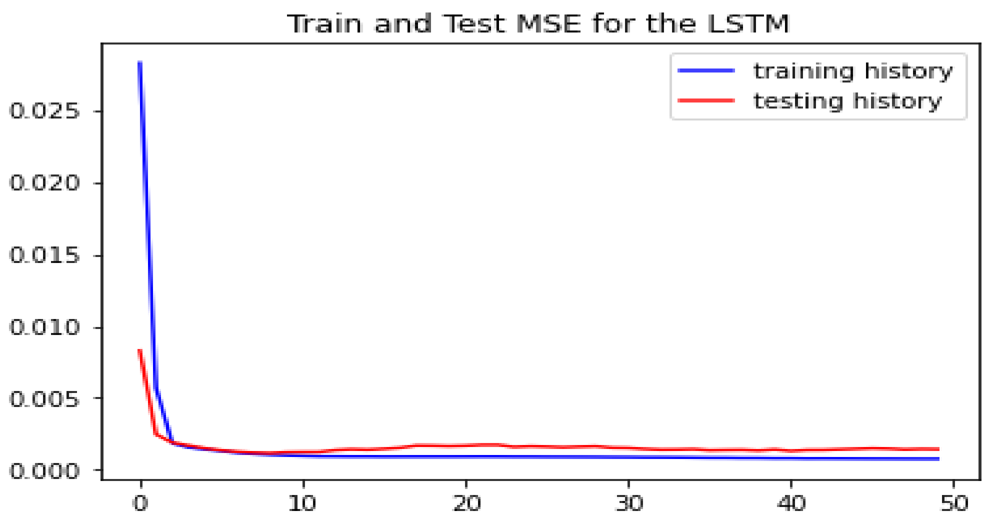


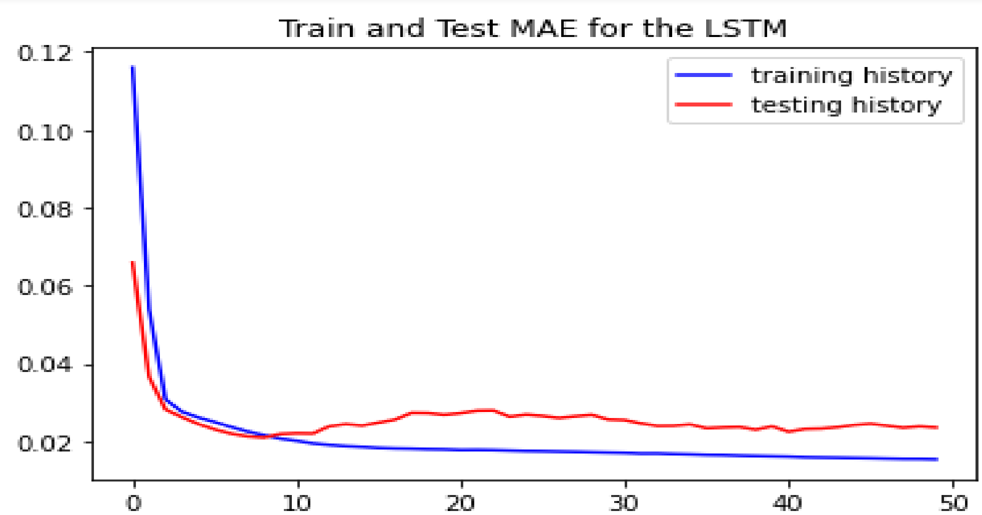


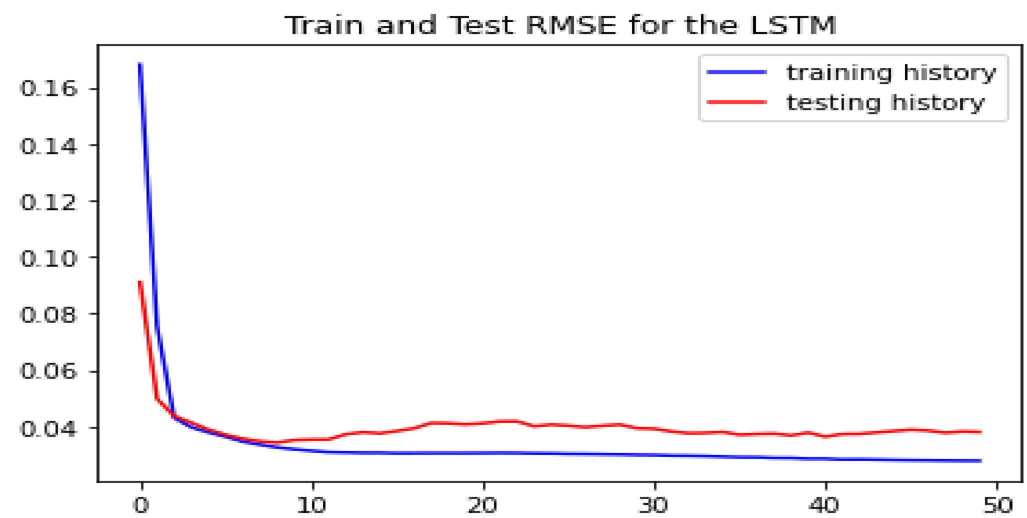
Test MSE:
Test RMSE:
Test MAE:

2. Bidirectional LSTM


Train and Test Loss for the LSTM
training history
testing history
0.12
0.10
0.08
0.06
0.04
0.02
0
10
20
30
40
50

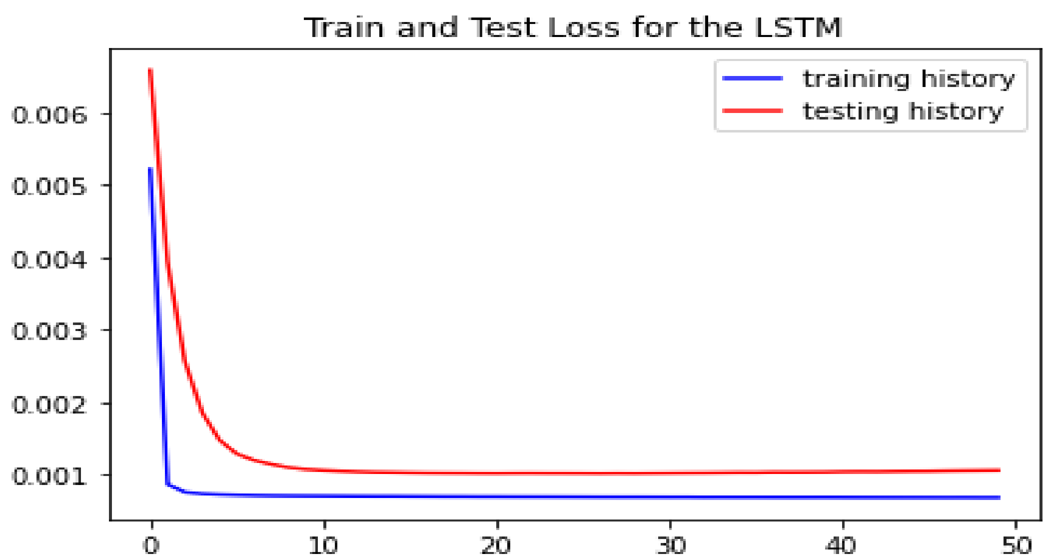

Train and Test MSE for the LSTM
training history
testing history
0.025
0.020
0.015
0.010
0.005
0.000
0
10
20
30
40
50

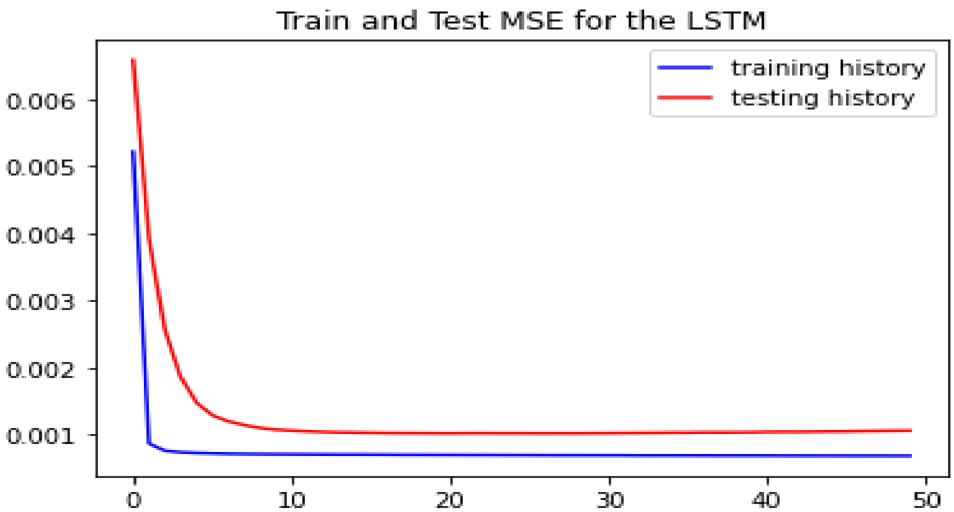

Train and Test MAE for the LSTM
training history
testing history
0.12
0.10
0.08
0.06
0.04
0.02
0
10
20
30
40
50

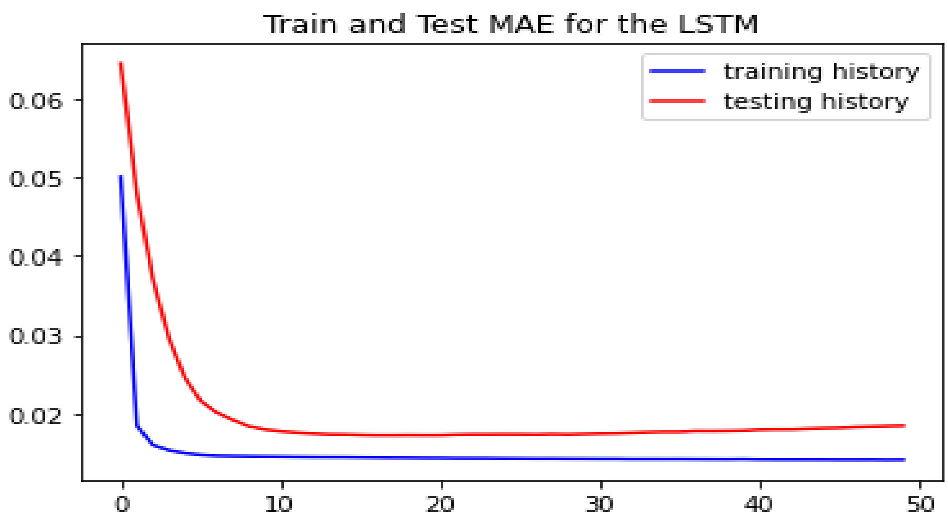


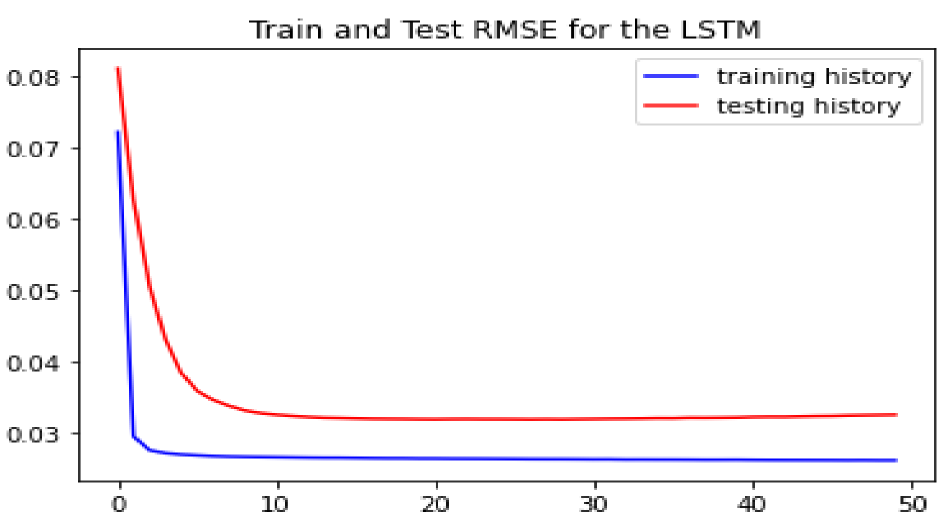
Test MSE: 475.559
Test RMSE: 21.807
Test MAE: 16.229

3. Results For Stacked LSTM




Train and Test MSE for the LSTM
training history
testing history
0.006
0.005
0.004
0.003
0.002
0.001
0
10
20
30
40
50


Train and Test MAE for the LSTM
training history
testing history
0.06
0.05
0.04
0.03
0.02
0
10
20
30
40
50

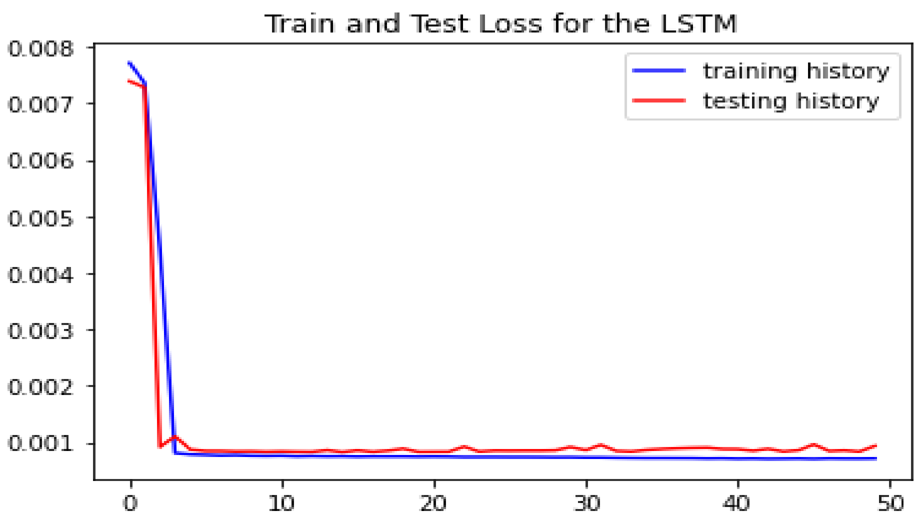

Train and Test RMSE for the LSTM
training history
testing history
0.08
0.07
0.06
0.05
0.04
0.03
0
10
20
30
40
50

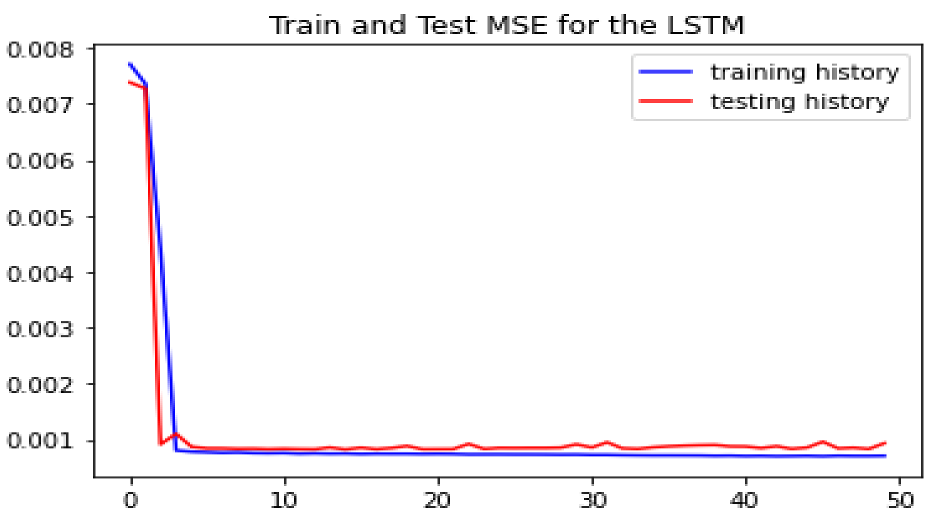
Test MSE: 232.004

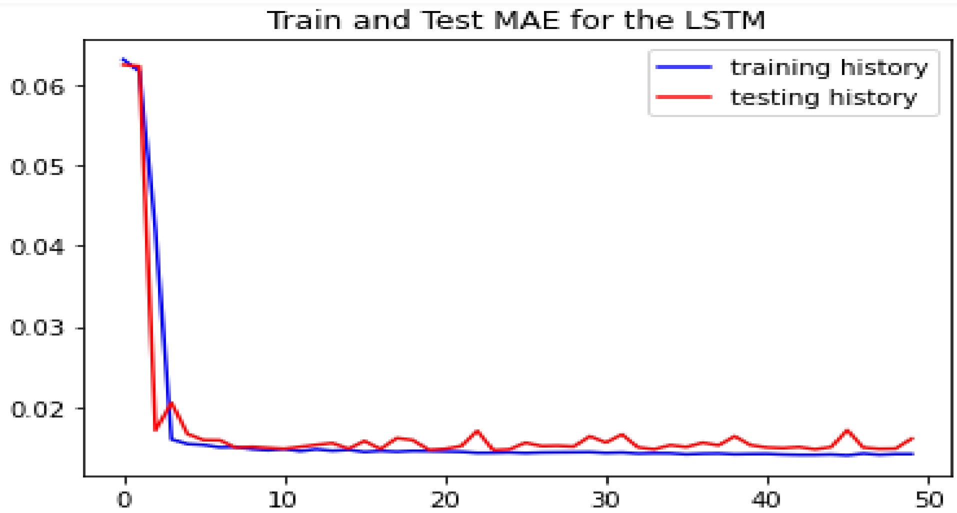
Test RMSE: 15.232

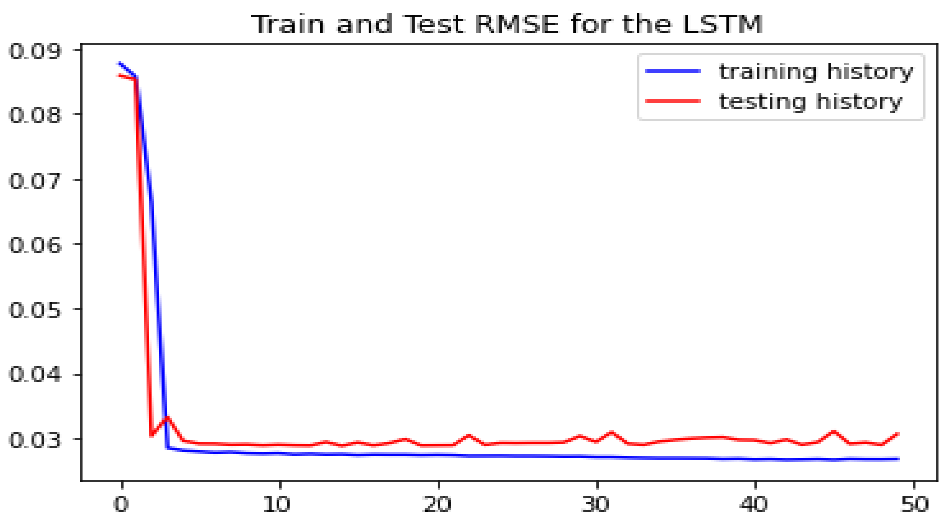
Test MAE: 12.262

4. Results For CNN LSTM









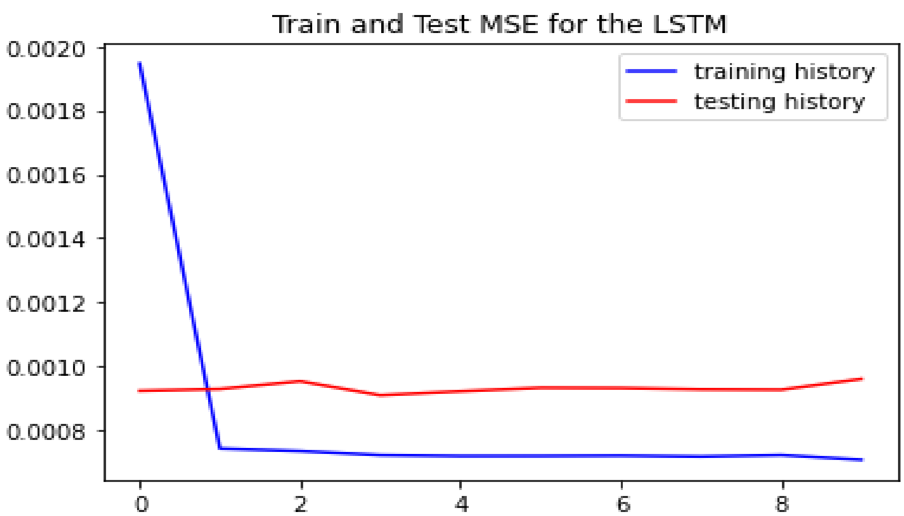
Test MSE:

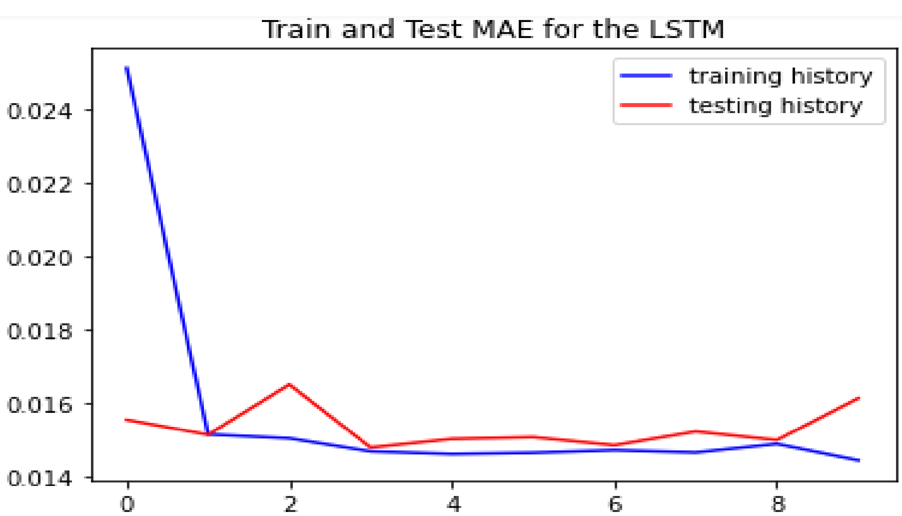
Test RMSE:

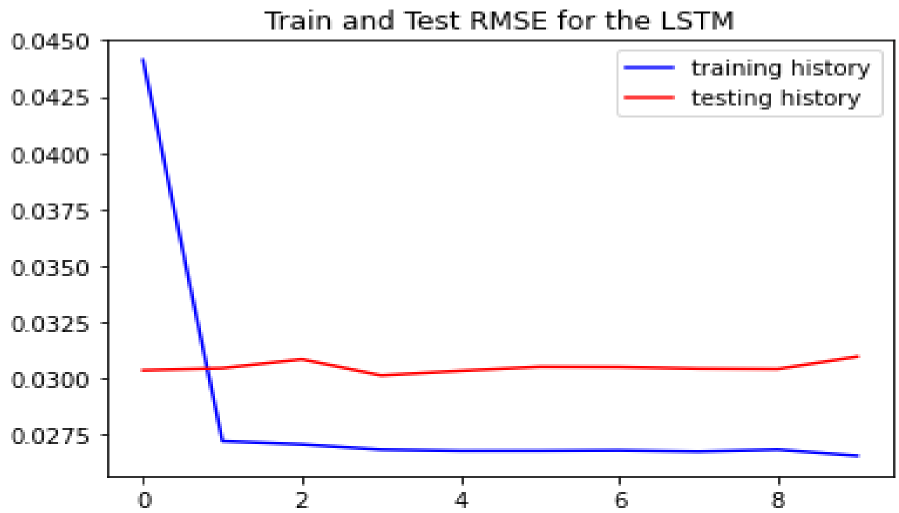
Test MAE:

5. Results For ConVLSTM


Train and Test Loss for the LSTM
training history
testing history
0.0020
0.0018
0.0016
0.0014
0.0012
0.0010
0.0008
0
2
4
6
8


Train and Test MSE for the LSTM
training history
testing history
0.0020
0.0018
0.0016
0.0014
0.0012
0.0010
0.0008
0
2
4
6
8


Train and Test MAE for the LSTM
training history
testing history
0.024
0.022
0.020
0.018
0.016
0.014
0
2
4
6
8



Test MSE:

Test RMSE:

Test MAE:

Table for showing different predicted errors for different LSTM models.

Model	loss	mse	mae	rmse
Vanilla LSTM	0.019	9.108	0.019	0.0302
Bidirectional LSTM	0.0188	8.997	0.0188	0.03
Stacked LStM	9.3976	9.3976	0.0177	0.0307
CNN LSTM	7.5166	7.5166	0.0138	0.0274
ConvLSTM	8.9152	8.9152	0.016	0.0299

We can see the different error predictions for different version of LSTM Models.

CONCLUSION

It is potential to accurately predict pollution, particularly by exploitation precise information from weather and pollution sensors from powerful sensors placed during a quiet atmosphere. we have a tendency to gift a way to require advantage of this correlation between sensors at totally {different|completely different} locations by combining predictions from different models into one with the utilization of Meta models. By coaching this LSTM correction, we've got considerably improved the prophetical ability, therefore accenting the link between totally different detector measurements, achieving RMSE results below twenty for many Meta models.
Because long-range prognostication tasks square measure inherently tougher, they need a lot of relevant historical information, together with optimum time intervals, that add another layer of optimisation to the LSTM model. this suggests that several hyperparameters, like cluster size and LSTM cell count, will still be optimized to come lower RMSEs for extended future predictions. The hope is that by exploitation the maximum amount statistic information as potential, we are able to produce stronger weights within the RNN, supported sequence dependencies. As mentioned higher than, longer forecast times will facilitate cities in philosophy and resource allocation, however a lot of significantly, will facilitate within the major battle against pollution to resolve this. downside for man and All living species on the planet forever.

REFERENCES

1.M. Zeinalnezhad, *Forecasting Air Pollution by Adaptive Neuro Fuzzy Inference System*, no. 2, pp. 2-4.

2.L. Mishra and S. Varma, Measurement of Air Quality Index using Internet of Things, pp. 1-5, 2019.

3.S. Dhingra, R. B. Madda, A. H. Gandomi, S. Member and R. Patan, *Internet of Things Mobile - Air Pollution Monitoring System (IoT-Mobair)*, vol. XX, no. Xx, 2019.

4.N. Kularatna, S. Member and B. H. Sudantha, *An Environmental Air Pollution Monitoring System Based on the IEEE 1451 Standard for Low Cost Requirements*, vol. 8, no. 4, pp. 415-422, 2008.

5.G. Lo Re, D. Peri and S. D. Vassallo, A mobile application for assessment of air pollution exposure.

6.H. Mokrani, R. Lounas and M. T. Bennai, Air Quality Monitoring Using IOT: A Survey, 2019.

7.S. Noureen, S. Atique, V. Roy and S. Bayne, "Analysis and application of seasonal ARIMA model in Energy Demand Forecasting: A case study of small scale agricultural load", *2019 IEEE 62nd Int. Midwest Symp. Circuits Syst.*, pp. 521-524, 2019.

8.Y. Zou, T. Wang, J. Xiao and X. Feng, *Temperature Prediction of Electrical Equipment Based on Autoregressive Integrated Moving Average Model*, no. 1, pp. 197-200, 2017.

9.Y. Pan, M. Zhang, Z. Chen, M. Zhou and Z. Zhang, An ARIMA Based Model for Forecasting the Patient Number of Epidemic Disease, pp. 31-34, 2015.

10.W. Wang, Air Pollution PM2. 5 Data Analysis In Los Angeles Long Beach With Seasonal ARIMA Model, pp. 2-5, 2009.

11.Q. Zhou, H. Jiang, J. Wang and J. Zhou, "Science of the Total Environment A hybrid model for PM 2. 5 forecasting based on ensemble empirical mode decomposition and a general regression neural network", *Sci. Total Environ.*, vol. 496, pp. 264-274, 2014.

12.J. B. Ordieres, E. P. Vergara, R. S. Capuz and R. E. Salazar, *Neural network prediction model for fine particulate matter (PM 2. 5) on the US e Mexico border in ‘rez (Chihuahua) El Paso (Texas) and Ciudad Jua*, vol. 20, pp. 547-559, 2005.

13.Y. Wang, J. Zhou, K. Chen, Y. Wang and L. Liu, Water Quality Prediction Method Based on LSTM Neural Network, pp. 5-9, 2017.

14.S. Alhirmizy, "Multivariate Time Series Forecasting with LSTM for Madrid Spain pollution", *2019 Int. Conf. Comput. Inf. Sci. Technol. Their Appl.*, pp. 1-5.

15.Y. Tsai, "Air pollution forecasting using RNN with LSTM", *2018 IEEE 16th Intl Conf Dependable Auton. Secur. Comput. 16th Intl Conf Pervasive Intell. Comput. 4th Intl Conf Big Data Intell. Comput. Cyber Sci. Technol. Congr.*, pp. 1074-1079, 2018.

16.Y. Jiao, Z. Wang and Y. Zhang, *Prediction of Air Quality Index Based on LSTM*, no. Itaic, pp. 17-20, 2019.

17.Xiaosong Zhao, Rui Zhang, Jheng-Long Wu, Pei-Chann Chang and Yuan Ze, "A Deep Recurrent Neural Network for Air Quality Classification", Journal of Information Hiding and Multimedia Signal Processing, 2018.

18.C R Aditya, Chandana R Deshmukh, D K Nayana and Praveen Gandhi Vidyavastu, "Detection and Prediction of Air Pollution using Machine Learning Models", International Journal of Engineering Trends and Technology (IJETT), 2018.

*19.*Gaganjot Kaur Kang, Jerry Zeyu Gao, Sen Chiao, Shengqiang Lu and Gang Xie, "Air Quality Prediction: Big Data and Machine Learning Approaches" in , International Journal of Environmental Science and Development, 2018.

20. Soubhik Mahanta, T. Ramakrishnudu, Rajat Raj Jha and Niraj Tailor, "Urban Air Quality Prediction Using Regression Analysis" in , IEEE, 2019.

21. https://www.geeksforgeeks.org/predicting-air-quality-index-using-python/

22. https://arxiv.org/abs/2102.10551

23. https://www.sciencedirect.com/science/article/pii/S1309104220301215

24. https://www.researchgate.net/publication/340636552_Air_Quality_Forecasting_using_LSTM_RNN_and_Wireless_Sensor_Networks

25. https://ieeexplore.ieee.org/document/8512020

26. https://towardsdatascience.com/forecasting-air-pollution-with-recurrent-neural-networks-ffb095763a5c?gi=54deb8972b3a

27. https://www.semanticscholar.org/paper/Air-Pollution-Forecasting-Using-RNN-with-LSTM-Tsai-Zeng/cb2e9d4ed7b36ce250de76c1f41234aa81c2abe0

28. https://machinelearningmastery.com/multivariate-time-series-forecasting-lstms-keras/

29. https://paperswithcode.com/paper/delhi-air-quality-prediction-using-lstm-deep

30. https://medium.com/analytics-vidhya/deep-learning-for-air-quality-prediction-b451e2936290