SHORT TERM AND LONG TERM WATER QUALITY PREDICTION USING MACHINE LEARNING

A PROJECT REPORT

Submitted by

BHAWNA BHATT [Reg No: RA1511008010238]
BANDARU TEJASWINI [Reg No: RA1511008010278]
HIMANSHI SHARMA [Reg No: RA1511008010282]

Under the Guidance of

Ms. K. Sornalakshmi
(Assistant Professor, Department of Information Technology)

In partial fulfillment of the Requirements for the Degree of

BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR – 603203

MAY 2019

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this project report titled "**SHORT TERM AND LONG TERM WATER QUALITY PREDICTION USING LSTM"** is the bonafide work of **"BHAWNA BHATT [Reg No: RA1511008010238], BANDARU TEJASWINI [Reg No: RA1511008010278], HIMANSHI SHARMA [Reg No: RA1511008010282],** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion for this or any other candidate.

Ms .K .SORNALAKSHMI
GUIDE
Assistant Professor
Dept. of Information Technology

Dr. G. VADIVU
HEAD OF THE DEPARTMENT
Dept. of Information Technology

Signature of Internal Examiner

Signature of External Examiner

ABSTRACT

Water is crucial for all types of life. The nature of water helps in controlling the biotic diversity, vitality, and rate of succession. The disintegrating nature of common water assets like lakes, streams, and estuaries is one of the direst and most troubling issues looked by humankind. The impacts of unclean water are broad, affecting each part of life. The security of water quality genuinely influences human wellbeing, fishery economy, and agrarian exercises. In this way, the administration of water assets is pivotal so as to upgrade the nature of water. The impacts of water pollution can be handled productively if the information is examined and water quality is anticipated in advance. As of late, water quality forecast has pulled in numerous considerations of governments and researchers. As the water nature of water assets become diverse and eutrophied, foreseeing and assessing water quality turns out to be increasingly vital. On the off chance that an early forecast of the water quality with a satisfactory precision can be accomplished, the negative effects will be limited or even be maintained a strategic distance from. The main objective of this study is to build up a water quality prediction model with the assistance of water quality parameters utilizing Long Short Term Memory (LSTM) Neural Networks and time series analysis. This study uses the water quality evident data of five and a half years from the distinctive zones of Georgia, United States of America. For this paper, data incorporates the estimation of 4 parameters which impact and affect water quality. To assess the performance of the developed model, the metrics used are Mean Squared Error, Root Mean Squared Error, and Regression Analysis.

ACKNOWLEDGEMENT

The success and the final outcome of this project required guidance and assistance from different sources and we feel extremely fortunate to have got this all along the completion of our project. Whatever we have done is largely due to such guidance and assistance and we would not forget to thank them. We express our sincere thanks to the Head of the Department, Department of Information Technology, Dr. G.Vadivu (PhD), for all the help and infrastructure provided to us to complete this project successfully and her valuable guidance. We owe our profound gratitude to our project guide Ms. K.Sornalakshmi (Assistant Professor), who took keen interest in our project and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system. We are thankful to and fortunate enough to get constant encouragement, support and guidance from all the Teaching staff of the Department of Information Technology which helped us in successfully completing our major project work. Also, we would like to extend our sincere regards to all the non-teaching staff of the department of Information Technology for their timely support.

BHAWNA BHATT

BANDARU TEJASWINI

HIMANSHI SHARMA

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 WATER QUALITY

Water assumes a crucial role in our day by day life and the nature of water in a region intensely influences the practical improvement of nearby ordinary industrial, agricultural and other anthropogenic activities. Common water resources like groundwater and surface water have dependably been the least expensive and most broadly accessible sources of freshwater. In any case, these assets are destined to progress toward becoming defiled because of different variables including human, industrial and commercial activities just as common procedures. Notwithstanding that, poor sanitation foundation and absence of mindfulness additionally contribute enormously to drinking water defilement. A considerable lot of the water pollutants have long haul negative effects on water quality, counseling a hazard to human wellbeing. Poor water quality influences the earth and human prosperity. Therefore, freshwater is seriously diminished. Additionally, contaminated water can prompt some waterborne ailments and furthermore impact child mortality. As indicated by the United Nations, waterborne infections cause the death of 1.5 million children for every year. The World Health Organization says that consistently more than 3.4 million individuals die because of water-related ailments. In this way, it is extremely essential to devise novel methodologies and techniques for deteriorating water quality and to figure future water quality patterns. So as to complete valuable and productive water quality analysis and foreseeing the water quality examples, it is important to incorporate a temporal dimension to the analysis, with the goal that the seasonal variation of water quality is tended to. Distinctive approaches have been proposed and applied for analysis and checking water quality and time series analysis.

1.2 PURPOSE

Predictive analysis can help to capture relationships among numerous variables that can help to assess risk with a particular set of conditions. The purpose is to propose an LSTM

NN, considering better and increasingly precise data to foresee and assess the water quality. This will help in foreseeing the estimations of water quality parameters dependent on their present qualities.

1.3 PROBLEM STATEMENT

There is an assortment of strategies utilized for water quality prediction at home and abroad. These strategies are principally separated into four classifications:-

1. Mathematical Statistics
2. Gray Theory
3. Chaos Theory
4. Neural Networks
5. ARIMA model

The strategy for mathematical statistical is powerful in modelling; however, the prediction is not perfect. The strategy for the gray theory is only appropriate for approximating exponential functions, not for the total non-linear functions. The strategy for chaos theory simply can be valuable when the training data is extremely well-off. The traditional neural network, whose advantages are non-linearity, self-organization learning is suitable for managing non-linear, arbitrary data. In light of the structure of the traditional neural network, it is not appropriate for dealing with the time series data. In spite of the fact that ARIMA models are very adaptable in speaking to various sorts of time series, AR, MA, and consolidated AR and MA(ARMA), their major impediment is the pre-assumed linear form of the model. In this way, no non-linear can be captured by the ARIMA model. Due to the few shortcomings from different models, LSTM model is picked so as to build up a far-reaching approach for effective water quality prediction and analysis.

1.4 OVERVIEW OF DATA

Past examinations have demonstrated that the extravagance and quality of data decide the exactness and unwavering quality of the analysis. Since the majority of the water

monitoring stations have an absence of detail and deficient observations, we have settled on the procurement of information from a standout amongst the most dependable water resources in the world. The sample data for this study has been procured from the United States Geological Survey's (USGS) National Water Information System(NWIS), which is an open data repository supporting the acquisition, processing and long haul storage of water quality over the United States of America. The measurements of the data have been gathered from different monitoring stations of Georgia and have been utilized for this study. Data from 1 October 2014 to 18 February 2019, with the time interval of 15 minutes has been obtained to do an effective prediction process utilizing this time series data that includes date/time, parameters and their measurements alongside their measurement units.

1.5 OBJECTIVES

The fundamental thought of this research is to devise a comprehensive methodology that analyzes and predicts the water nature of specific regions with the assistance of certain water quality parameters. These parameters incorporate physical, biological and chemical factors which impact water quality. This research expects to address this issue by recommending a model dependent on Machine Learning procedures so as to anticipate the future water quality patterns of a specific region with the assistance of historical water quality data. LSTM model is utilized to build up a methodology for viable water quality forecast and analysis. The model based on LSTM NN for water quality forecast is contrasted with two models: one with the traditional neural network and other based on ARIMA model. The results verify the efficiency of the model we have proposed.

1.6 ORGANIZATION OF THE REPORT

The report has been divided into the following sections:

1. Section two describes the related work done using different approaches.
2. Section three focuses on the background and strategies of this research, it introduces LSTM, ANN and ARIMA and discusses how an LSTM model is used in this prediction.
3. Section four focuses on the application problem domain and study area.

4. Section five presents experimental results and performance evaluation of the model.
5. The final section summarizes the results of the research and concludes some directions for future work.

CHAPTER 2

LITERATURE STUDY

Water quality prediction provides a significant reference for dynamic regulation of water quality and sudden events. Water quality forecast has increasingly pragmatic noteworthiness for the administration of water assets as well as for the avoidance of water contamination. It's a period arrangement forecast issue which the conventional neural system isn't reasonable. In the paper[1] new water quality forecast strategy dependent on long and short memory neural system (LSTM NN) using neural networking for water quality expectation is proposed. Initially, an expectation display dependent on LSTM NN is set up. Furthermore, as the preparation information, the informational collection of water quality markers in Taihu Lake which estimated month to month from 2000 to 2006 years is utilized for the preparation display. Thirdly, to improve the prescient exactness of the model, a progression of reproductions and parameters determination is completed. In the recent years, researchers have been endeavoring to utilize profound learning-based techniques to take care of time arrangement expectation issue. The long and short memory neural system (LSTM NN) has 'memory' as its very own result one of a kind system structure.

With nonlinear enactment capacities, neural systems are approximations to nonlinear capacities. The main drawback of BPNN of easily falling into local minima in the training process is overcome by LSTM & RNN. RNNs and LSTMs are accordingly basically a nonlinear time arrangement display, where the nonlinearity is found out from the information. These won't do well with little measures of information since it needs to gain proficiency with the nonlinearity. At the point when the change of information is little, the prescient precision of BP NN is higher than OS-ELM. At the point when the change of information is extensive, the prescient exactness of OS-ELM is higher than BP NN.

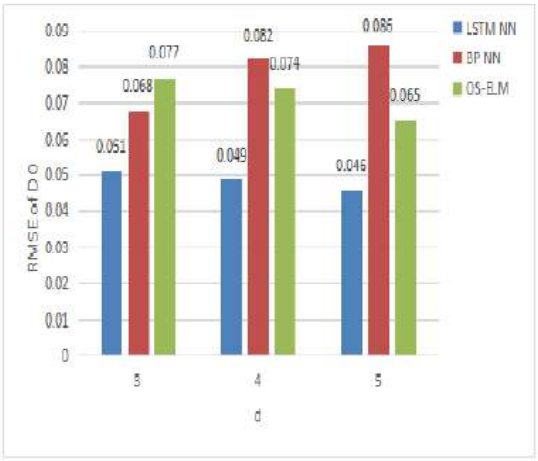
LSTM has been connected in the field of time forecast effectively, for example, stock expectation [2] and traffic stream [3] expectation. A forecast model presented incorporates the input layer, hidden layer and output layer. Also, the model is prepared by recorded water quality parameters. Thirdly, the prescient precision is improved by parameters choice and a progression of recreations. At long last, the technique dependent on LSTMNN for water

quality forecast is contrasted and two strategies: one depends on back spread neural system (BP NN), the other depends on online successive outrageous learning machine (OS-ELM).

The performance metric show the relevant and effective accuracy of this model .

TABLE I. THE SIMULATION RESULTS OF THE PROPOSED MODEL

			DO		TP	
d	*Hiddnum*	*epoch*	*RMSE*	*MaxError*	*RMSE*	*MaxError*
3	10	10	0.066	0.160	0.043	0.116
	10	20	0.061	0.156	0.048	0.155
	13	25	0.058	0.151	0.042	0.118
	15	**20**	0.051	0.138	**0.041**	**0.103**
	15	25	0.102	0.215	0.054	0.188
	17	30	0.060	0.225	0.043	0.094
4	10	20	0.053	0.152	0.044	0.118
	13	25	0.054	0.162	0.047	0.135
	15	20	0.049	0.127	0.042	0.130
	15	25	0.050	0.181	0.046	0.108
	17	30	0.054	0.153	0.046	0.097
5	10	20	0.057	0.198	0.043	0.112
	13	25	0.061	0.196	0.041	0.121
	15	20	0.053	0.143	0.042	0.103
	15	**25**	**0.046**	**0.116**	0.047	0.125
	17	30	0.048	0.127	0.043	0.124



In TABLE I, MaxError is the most extreme error between the estimation of forecast and genuine information. As can be seen from TABLE I, RMSE bit by bit diminishes as Hiddnum increments, at the point when Hiddnum is more prominent than specific esteem, RMSE starts to increment. In a similar estimation of Hiddnum, as the age builds, RMSE steadily diminishes, when age surpasses specific esteem, RMSE starts to increment. At the point when the vacillation of information is little, the prescient exactness of BP NN is higher than OS-ELM. At the point when the vacillation of information is expansive, the prescient precision of OS-ELM is higher than BP NN.

Improving the precision of the water quality expectation is an essential and troublesome assignment confronting leaders in water assets the executives. Numerous analysts have contended that consolidating diverse models can be a successful method for improving their prescient execution. The hybrid models[4] of an autoregressive coordinated moving normal (ARIMA) and neural system, as a standout amongst the most prominent crossover models for time arrangement estimating, have as of late been appeared for water quality expectation. Nonetheless, these models have numerous suspicions and impediments. This paper, a novel hybrid model of ARIMA and Radial Basis Function Neural Network(RBF-NN) is proposed

so as to yield more broad and higher exactness forecast show than conventional crossover ARIMA-ANNs models for water quality expectation. ARIMA which is a linear model and used to obtain the existing linear structure and RBF-NN used to capture the non-linear architecture and then perform the predictions.

This paper actually depicts the real time problems with the existing famous forecasting model ARIMA which basically cannot the handles the large datasets and non-linear pattern . In the ARIMA-RBF model, the usage of unique capability of ARIMA model to identify and magnify the existing linear component, and then an RBF neural network is used to capture the underlying data generating process and predict the future, using the residuals from the linear model, the observed data and the predicted value of the ARIMA model.

The paper have majorly two major assumptions which effects the performance of the model:-

- The connection among linear and non-linear components are added and along with these there will be chances of underestimation of the connection between the parts and finally corrupts execution.
- One may not ensure that the residuals of the straight part may include a substantial non-direct example.

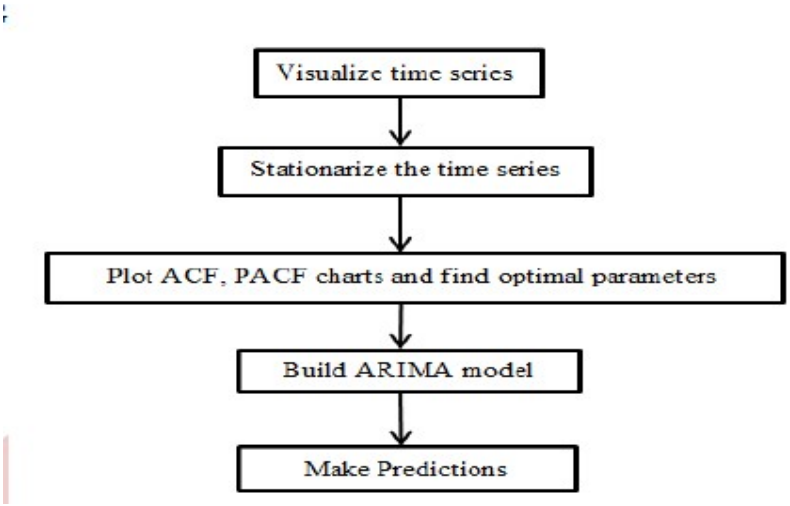
PERFORMANCE METRICS

Table II Statistical comparison of observed and predicted data from the proposed model and other three modes

Model	DO			NH3-N		
	R	MAPE(%)	RMSE	R	MAPE(%)	RMSE
ARIMA	0.886	4.59	0.6194	0.8350	17.03	0.0502
RBF-NN	0.8809	4.94	0.6316	0.8516	15.31	0.0481
Zhang's model	0.8928	4.48	0.6045	0.8696	14.09	0.0460
Our proposed model	**0.9013**	**4.30**	**0.5735**	**0.8781**	**13.30**	**0.0445**

Here there are three values as the evaluation measures R, MAPE and RMSE for the models used in the paper. On account of the Pearson product moment correlation coefficient (R) among observed and anticipated information, the improvement of the proposed novel model over the ARIMA show, the RBF-NN display, the Zhang's model were 1.53%,2.04%, 0.85% for DO and 4.31 %, 2.65%, 0.85% for NH3N, individually. There was an abatement of 0.29%, 0.64%, 0.18% in the RMSE esteems for DO and 3.73%, 2.01%, 0.79% for NH3N.

Stream contamination is a another major issue in India as well as over the world for mankind and all the living species on Earth. As indicated by the Human Development Report 2006 of WBCSD, at present, nearly 4 billion individuals live in stream bowls where the use of water surpasses least revive levels, driving towards the absence of waterways and decrease of groundwater. Over 70% of modern waste is untreatably dumped into the water. In this paper[5] they have dissected and estimated water quality parameters like Temperature, pH, Turbidity, conductivity, broke down oxygen for stream utilizing time series analysis and ARIMA displaying a measurable examination which is extremely useful in long range to screen the soundness of the waterway.



This paper has designed the simple ARIMA forecasting model using the dataset of water quality parameters of River Burnett (2015).

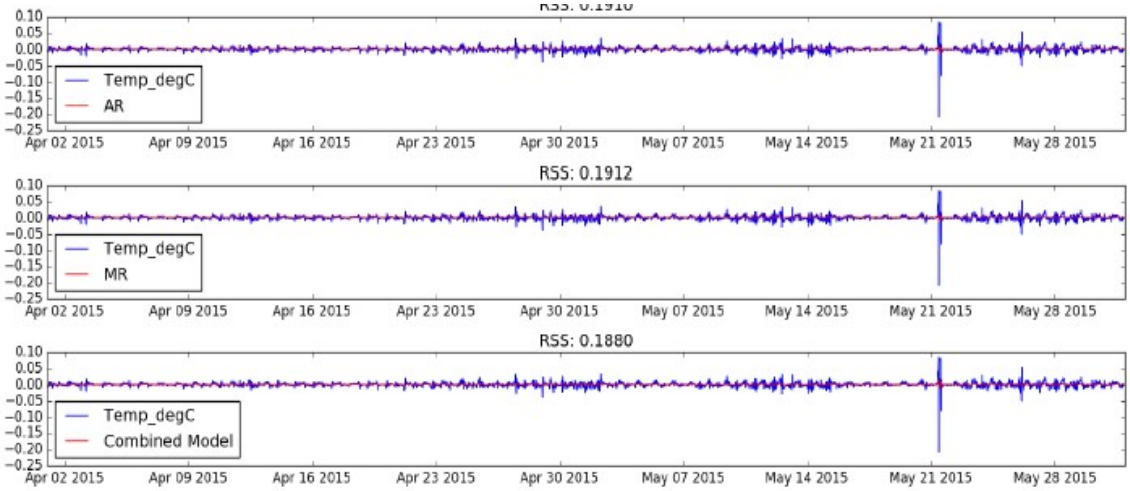


Fig 2.AR, MA and Combined Model for Temperature (April – May 2015)

AR, MA and Combined model with its total squared residual for forecasting depicts the relevant RSS values for the temperature for one month .For short datasets the results shows that the ARIMA is best fitted model in forecasting. Autoregressive Integrated Moving Average model is a short-term (at least 40 data points) time-series prediction model which can be used primarily for predicting data which has a low variance or fewer outliers and tends to follow a stable trend. This model is most suited for data which shows a high-level of seasonality. One of the least complex and compelling strategies which portray the patterns, regularity, abnormality and cyclic conduct of the datasets. Despite the fact that ARIMA models are very adaptable in that they can speak to a few unique sorts of time series, e.g., pure autoregressive (AR), pure moving normal (MA) and combined AR and MA (ARMA) series, their significant confinement is the pre-accepted direct type of the model. Therefore, no nonlinear patterns can be caught by the ARIMA models. To account the limitation of linear models and to represent certain nonlinear patterns saw in a genuine issue, a few classes of nonlinear models have been proposed in the writing. These incorporate the limit autoregressive (TAR) show and the autoregressive restrictive heteroscedastic (ARCH) demonstrate. Since these nonlinear models referenced above are created for explicit nonlinear patterns, they are not fit for displaying different kinds of nonlinearity in time series.

The security of water quality genuinely influences human wellbeing, fishery economy and agrarian exercises. On the off chance that an early expectation of the water quality with an adequate exactness can be accomplished, the negative effects will be limited or even be kept away from. Numerous scientists have connected neural systems (ANNs) to assemble the water quality models for there is a confused non-linear relationship between the forecast factors and estimated inputs. In any case, the ANN models are easy but difficult to be over-fitting for preparing them needs substantial tests. As the target of the examination, this paper [6]utilized neural system and Markov chain approach to build up another crossbreed strategy for anticipating the biochemical oxygen demand (BOD) which is the fundamental marker of water quality. ANN produces the essential qualities and after that, the outcomes are changed by three relapse strategies utilizing the Markov transitional likelihood networks individually. Here, they have utilized a 27-year water

quality informational collection of Tolo Harbor which just have an aggregate of 439 examples to test the technique.

MODELS	RMSE	Bias	R^2
ANN alone	0.6597	-0.13032	0.8610
ANN-Markov chain-linear	0.6504	-0.0954	0.8649
ANN-Markov chain-ANN	0.6311	-0.0974	0.8727
ANN-Markov chain-SVR	0.6174	-0.0426	0.8782

Table 2. PERFORMANCE PARAMETERS OF THE PREDICTION MODELS

The RMSE, Bias and R2 are displayed in Table. Can be concluded that the ANN-Markov chain models show preferable execution over the ANN display alone in all measure pointers, and the ANN-Markov chain demonstrates utilizing SVR is the most fitting technique with RMSE of 0.6174, Bias of - 0.0426 and R2 of 0.8782. The ANN model used for training the primary models to generate basic prediction values and Markov chain used for the transition probability matrix calculations between the different states of predicted values.ANN is well-suited method with self adaptability, self organization and error tolerance which is quite suitable for non-linear simulations. Though it utilizes 351 records of water quality parameters of TM3 stations to foresee essential qualities but in the BOD calculations there are lot more uncertainties and big challenges of possibility of over fitting, as it uses large amount of training datasets and required many iterations and the outcomes in the testing dataset are not all around fit with deliberate qualities.

Another paper[7] which exhibits Classification information demonstrate utilizing decision tree to analyze water quality information of MAA Narmada River at Harda locale. The information demonstrate was executed in WEKA programming. Arrangement utilizing choice tree was connected to order/foresee the poison class of water. It is seen in the investigation that the Nitrogen(NH3_N, NO3_N), pH, Temp _C, BOD, COD, other parameters pertinent to water forms assume a vital job to evaluate the nature of waterway water. In this investigation, we have utilized five traits of water quality information which can influence the precision of water.

Year	PH_gen	DO	BOD	No3_N	NH3_N	Class
1990	8.2	8.1	0.8	0.39	0.08	I
1991	7.2	8.3	0.9	0.52	0.02	II
1992	7.9	0.9	1.4	0.41	0.03	I
1993	8.2	7.8	1.8	0.55	0.02	III
1994	8.3	7.9	1.3	0.56	0.01	IV
1995	8.3	8.1	1.9	0.58	0.03	III
1996	8.9	8.9	1.3	1.44	0.03	II
1997	8	7.5	0.9	0.47	0.02	I
1998	8.3	8.3	0.5	0.51	0.12	IV
1999	8.1	7.9	0.9	0.25	0.05	I
2000	8.4	8.2	0.6	0.67	0.03	II
2001	8.3	8.1	1.5	0.44	0.03	IV
2002	8.3	7.8	1	0.28	0.01	IV
2003	8.4	7.7	1.3	0.51	0.02	II
2004	8.1	8.1	0.6	0.52	0.03	II
2005	8.2	7.5	1.1	0.43	0.02	IV
2006	8.2	8.1	1.3	0.06	0.01	IV
2007	7.9	7.6	0.8	0.49	0.02	II
2008	8.3	7.3	1.5	0.35	0.03	IV
2009	8.3	7.2	1.2	0.51	0.12	IV
2010	8.4	7.4	1.2	0.31	0.13	IV
2011	8.1	7.3	1.3	0.85	0.02	III
2012	8.0	8.2	1.3	0.56	0.12	III

Wator Quality Parameter	Class I	II	III	IV
PH (mg/l)	<5	5-9	5-9	>9
DO(mg/l)	>6	6	4	<2
BOD(mg/l)	<1.5	1.5	2	>4
No3_N(mg/l)	<5	5	5	>5
NH3_N(mg/l)	<0.5	0.5	0.5	0.5

Table.3 Water quality dataset after pre-processing

The decision tree offers numerous advantages to data mining like is straightforward by the end client , can deal with an assortment of information: Nominal, Numeric and Textual, able to process mistaken datasets or missing qualities, high execution with few endeavors. An informational collection gathered isn't legitimately appropriate for enlistment (learning obtaining), it involves by and large commotion, missing qualities, and conflicting informational collection is excessively huge, etc.

The correctly classification of instances was 95.4545% and incorrectly classification is 4.5455%. Metrics depicts that if pH esteem goes to not exactly or equivalent to 8.1 unit and the measure of NO3 N increment then the nature of surface water decline at one class level. That implies the measure of NO3 N expands causes increment contamination in surface water. On the off chance that pH esteem goes to more prominent than 8.1 and the measure of NO3-N lies somewhere in the range of 0.47 and 0.51 at that point decline in the sum in BOD likewise Decrease the nature of water at one dimension class. While as increment the incentive in NO3-N and DO in surface water, the nature of water additionally improved at one dimension of class.

Another paper[8] which utilized ARIMA Model (2.0.0) to figure the pattern of changes in TSS, DO and NO3 parameters of two stations of Sefidrud River in Astaneh and Manjilin measurable year of 2010 independently at each station. The relapse coefficient of real and fitted values for every parameter was 85%, 74% and 75% in Astaneh station

and 78%, 80% and 82% in Manjil the station, individually. To test the gauging directed, they additionally utilized the information of the year 2011 for Astaneh station and information of the year 2012 for Manjil station, which stressed on a decent and generally great execution of the exhibited model. This paper though used classification technique but this approach has many limitations like this will not effectively work the case for pure numeric datasets and is not effectively gives the information of relationship between the dependent and independent indicators of the water quality.

CHAPTER 3

PROPOSED METHODOLOGY

3.1 **THEORETICAL BACKGROUND**

Machine Learning is the use of Artificial Intelligence (AI) that gives systems the capacity to consequently learn and improve from experience without being unequivocally programmed. It centers around the development of computer programs that can get the data and use it for learning themselves.

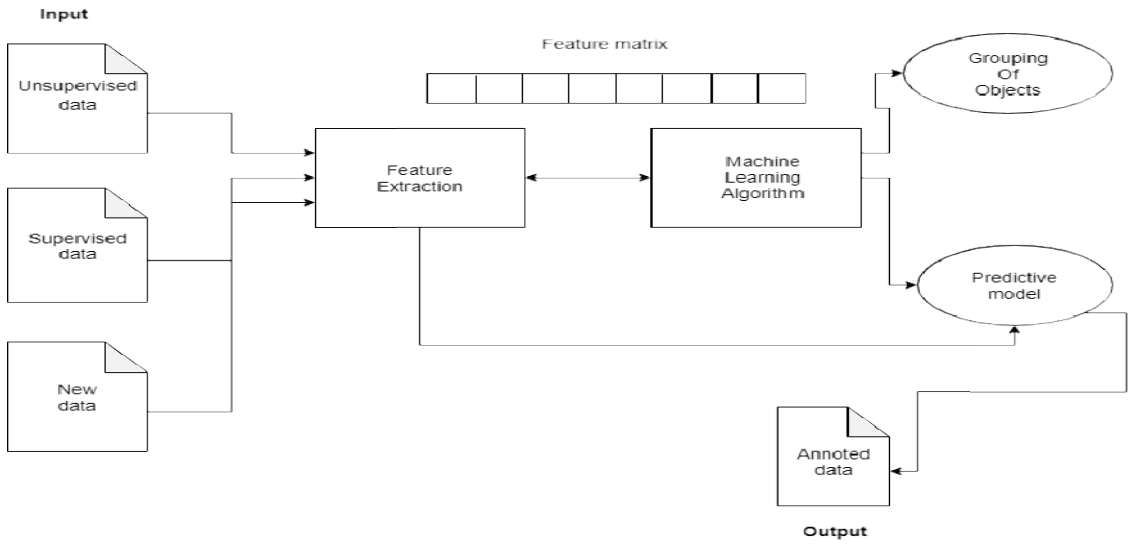


Figure 3.1:- Applying machine learning algorithm on the training dataset to train the model

The process of learning starts with observations or information, such as examples, direct expertise, or instruction, so as to look for patterns in data and settle on better choices later on based on the examples we give. The main aim is to enable the computers to adapt consequently without human interference or help and alter the activities as required.

3.1.1 **TIME SERIES FORECASTING**

A time series is a series of data points listed in time order. A time series is a sequence at successive equally spaced points in time. It is a sequence of discrete-time data. It is a set of observations, x_t, each one being recorded at a specific time. A discrete time series is one in which the set T_0 of times at which observations made is a discrete set. Continuous time series is obtained when observations are recorded continuously over some time interval, for example, when $T_0 = [0,1]$.

Time series analysis involves techniques for studying time series data so as to obtain meaningful statistics and different characteristics of the data. Time series forecasting is the utilization of a model to predict future values based on historical observed data.

The methodology utilized in this research involves Machine Learning with training and testing data from USGS online data repository. The theoretical background of the methodology is as follows:

3.1.1.1 **LSTM**

Recurrent neural networks (RNN) are networks with loops in them, enabling the information to persevere. When the gap between the related information and the place it is required is small, RNNs can learn to utilize the past information. Unfortunately, as the gap increases, RNNs become unfit to learn to associate the information.

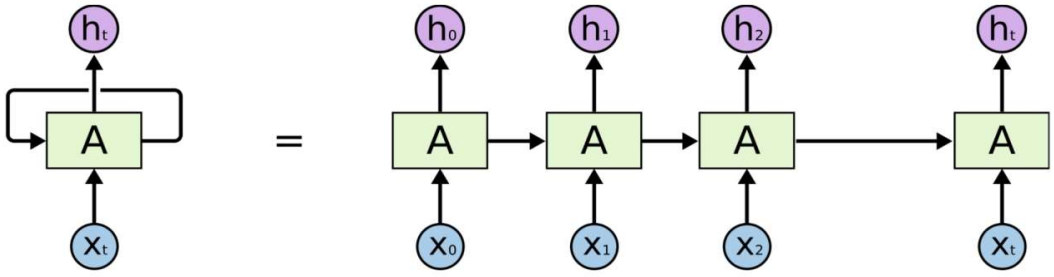


Figure 3.2:- Recurrent Neural Network

LSTMs are an extraordinary sort of RNNs, equipped for adapting long term conditions. Recollecting information for long periods purposes their default behaviour. LSTMs also have a chain like structure, yet the repeating module has an alternate structure, not at all like RNNs. Rather than having a single neural network, there are four layers, cooperating in a unique manner. The way to LSTM is the cell state. The cell state is somewhat similar to a conveyor belt. It runs straight down the whole chain, with some minor linear connections. It is extremely simple for information to the cell state, carefully controlled by structures called gates.

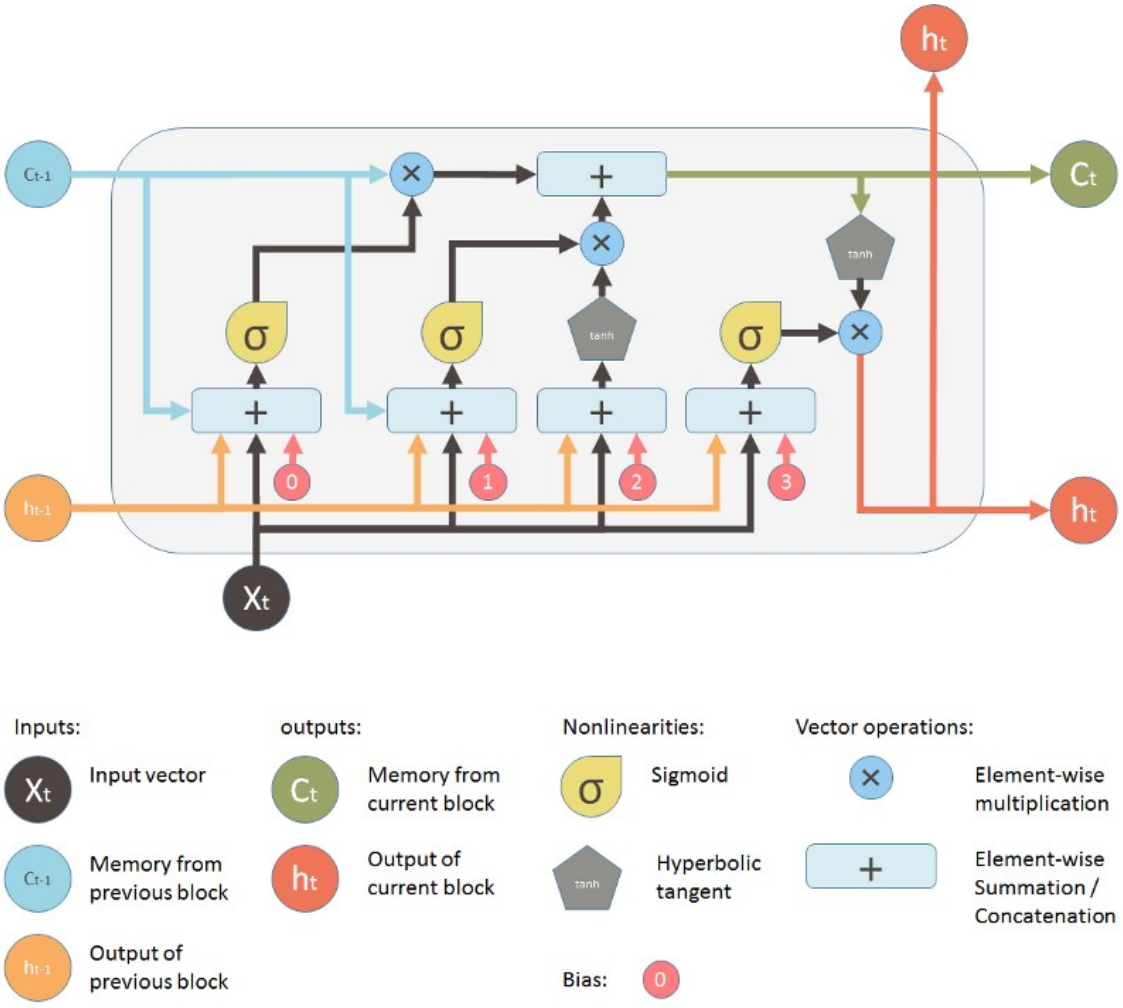


Figure 1.3:- Basic LSTM Memory Block

LSTM networks are appropriate for classifying, processing and making predictions based on time series data, since there can be lags of obscure duration between important events in a time series. They were created to manage the exploding gradient and vanishing gradient

problems that can be experienced when training traditional RNNs. The activation function of the LSTM gates is frequently the logistic function. The weight of these connections, which need to be learned during training, decide how the gates operate.

A RNN utilizing LSTM can be trained in a supervised fashion, on a set of training sequences, using an optimization algorithm, gradient descent, joined with back propagation through time to calculate the gradients needed during the optimization process, in order to change weights.

Gates are an approach to alternatively let data through. They are made out of a sigmoid neural net layer and a point wise multiplication operation.

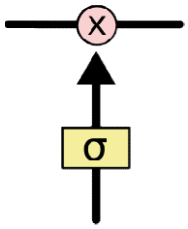


Figure 3.2:- Block diagram of a gate

The sigmoid layer yields numbers somewhere in the range of 0 and 1, depicting the amount of every component ought to be let through. A value of 0 signifies "let nothing through", while a value of 1 signifies "let everything through". An LSTM has three of these gates, to secure and control the cell state.

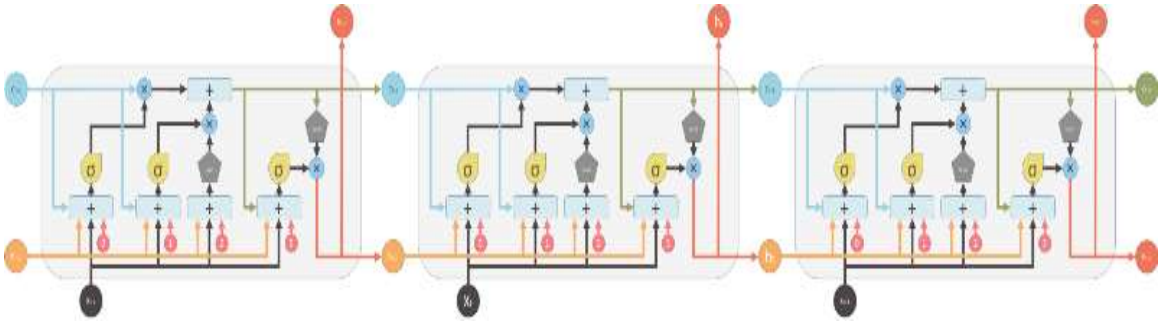


Figure 3.3:- LSTM network with memory blocks

The initial phase in our LSTM is to choose what data we are going to discard from the cell state. This choice is made by the sigmoid layer, called the "forget gate" layer. It looks at h_{t-1} and x_t and yields a number somewhere in the range of 0 and 1 for every cell state C_{t-1}. 1 signifies "totally keep this" and 0 implies “totally dispose of this".

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f) \quad (3.1.1.1.1)$$

f_t is the value of the forget gate at t^{th} time. W_f is the weight between the forget gate and the input layer. h_{t-1} is the output of the previous memory block. x_t is the input vector. b_f is the bias vector.

Following stage is to choose what new data we are going to store in the cell state. This has two sections. Initial, a sigmoid layer, called the input gate layer, chooses which values should be updated. Next, a tanh layer makes a vector of new candidate values, C't, which could be added to the state.

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \quad (3.1.1.1.2)$$

$$C'_t = \tanh(W_c.[h_{t-1}, x_t], b_c) \quad (3.1.1.1.3)$$

i_t is the value of the input gate. W_i is the weight between the input gate and the input layer. b_i is the bias vector. W_c is the weight between the input gate and the hidden layer.

After that, we update the old cell state, C_{t-1}, into the new cell state, C_t. We multiply the old state by f_t, overlooking the things we chose to overlook before. Then we add $i_t * C'_t$. This is the new candidate values, scaled by the amount we chose to update each state value.

$$C_t = f_t * C_{t-1} + i_t * C'_t \quad (3.1.1.1.4)$$

C_t is the memory from the current block. C_{t-1} is the memory of the previous block.

At last, we have to choose what we are going to yield. This output will be based on our cell state however will be a filtered form. To begin with, we run a sigmoid layer which chooses what parts of the cell state, we are going to yield. At that point, we put the cell state through tanh (to push the values in the range of - 1 and 1) and multiply it by the result of the sigmoid gate, with the goal that we just output the parts we chose to.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o \quad (3.1.1.1.5)$$

$$h_t = o_t * \tanh(C_t) \quad (3.1.1.1.6)$$

o_t is the value of the output gate. W_o is the weight between the hidden layer and the output gate. b_o is the bias vector. h_t is the output of the current block.

Thus, this single unit settles on choice by thinking about the present information, past output and past memory. What's more, it produces new output and adjusts its memory.

3.1.1.2. **ANN**

ANN is a computational strategy, motivated by the researches of the brain and sensory systems in natural life forms. It speaks to very deified mathematical models of our present interpretation of such complex frameworks. One of the qualities of the neural systems is their capacity to learn.

A usual neural network consists of an input layer, which gets inputs of the model and calculates the weighted sum of inputs, an output layer, which gives the final outputs and one or more intermediary hidden layers for processing. Every layer has one or more handling nodes called neurons and each neuron has its own transfer function. In addition to that, there are interactions between the model with changing "weights".

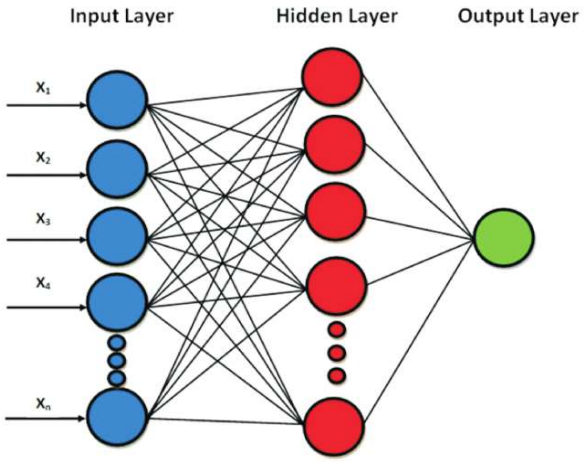


Figure 3.4:- An ANN model with three layers

The most generally utilized neural network is the three-layer feed-forward neural networks with backpropagation. The customary back propagation network depends on a gradient descent algorithm, which circulates the output error through the model after one cycle or "epoch" in a back direction. The weights will be balanced amid the algorithm to understand

the minimum error. The proper number of neurons in the hidden layer and the transfer functions are selected by trial and error methods, just as relevant experience. In the feedforward procedure, the weights are multiplied by the inputs and the resultant value is pushed ahead towards the following layer, until it achieves the output layer, as follows:

$$z_i = \sum_{j=i}^{m} w_{ij}.x_{ij} \quad (3.1.1.2.1)$$

Here, w_{ij} is the weight transferred from the j^{th} input to the i^{th} node, is the input. z_i is the summation of inputs of the i^{th} node.

The backpropagation procedure decides the error value by computing the distinction between the predicted value and expected value, beginning from an output layer towards the input layer. It is indicated by the symbol $\delta(l)j$, which is equivalent to the error of node j in layer l.

$$\delta(l)j = z_j - y_j \quad (3.1.1.2.2)$$

It is a repetitive process so after modifications of the weights, the procedure is run over and again until convergence.

An ANN stores the information about the issues as far as weights of interconnections. The way toward deciding ANN weights is called learning and training. The ANNs are prepared with a training set of input and known output information. The number of input nodes, output nodes in the hidden layer relies upon the issue being considered. In the event that the number of nodes in the hidden layer is little, the system might not have adequate degrees of freedom to gain proficiency with the procedure effectively. If the number of nodes is high, the training will take some more time and the network may sometimes overfit the data. Subsequent to training is finished, the ANN execution is checked. Contingent upon the result, either the ANN must be re-trained or it very well may be executed for its planned use.

3.1.1.3 **ARIMA**

Auto-Regressive Integrated Moving Average (ARIMA) is a class of factual models for analyzing and predicting time series data. It is a speculation of the less complex Auto-Regressive Moving Average (ARMA) and includes the thought of integration. ARIMA models are applied where data show proof of non-stationarity.

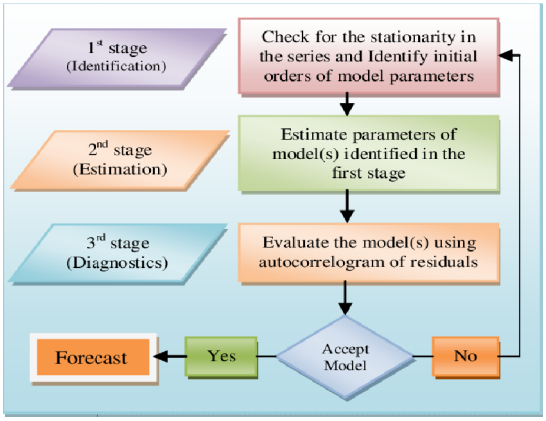


Figure 3.5:- Flow chart of ARIMA model

The AR part of ARIMA shows that the advancing variable of interest is relapsed on its lagged values. The MA part demonstrates that the regression error is really a direct combination of the error terms whose values came contemporaneously and at different times in the past. The I shows that the data values have been supplanted with the difference between their values and the past values. This differencing procedure may have been executed more than once. The reason for each one of these features is to make the model fit the information just as conceivable.

Non-seasonal ARIMA models are usually indicated by ARIMA (p,d,q) where parameters

1. p - number of lag observations added in the model, also known as the lag order.
2. d - number of times that the raw observations are differenced, also known as the degree of differencing.
3. q - size of the moving average window, also known as the order of moving average.

The autoregressive (AR) part of the model is explained by the following equation:

$$Y_t = C + \emptyset_1 Y_{t-1} + \emptyset_2 Y_{t-2} + \cdots + \emptyset_p Y_{t-p} + e_t \qquad (3.1.1.3.1)$$

The moving average (MA) part of the model is explained by the following equation:

$$Y_t = C + \emptyset_1 e_{t-1} + \emptyset_2 e_{t-2} + \cdots + \emptyset_p e_{t-p} + e_t \qquad (3.1.1.3.2)$$

CHAPTER 4

SYSTEM ANALYSIS

It is critical to characterize functional and non-functional necessities when fabricating a water quality prediction model for ensuring that the model we assemble will enable clients to accomplish the business targets. Useful necessities characterize what the framework does. They contain the highlights that the information distribution centre framework ought to have. Non-functional prerequisites direct and oblige the design. The functional requirement is like describing the characteristics of the system as it relates to the system's functionality. The non-functional requirement emphasises on the performance characteristic of the system.

4.1 **FUNCTIONAL REQUIREMENTS**.

Interoperability / Open Architecture: There is no standard or uniform foundation stage. The key thought is whether the investigation arrangement or analytics solutions works with numerous stages or is a shut extra to one stage. Here we will be designing open source model which is platform independent.

Machine Learning Methodology: Each Predictive Asset Maintenance arrangement depends on a Big Data approach. Is this a manual procedure or is Artificial Intelligence used to consequently choose the ideal calculation for the particular situation.

Resource Visualization: At a facility level, professionals getting to the UI won't be prepared in Artificial Intelligence and Big Data. The key contemplations when characterizing this necessity are the representation of machine conduct and the capacity to delineate the strength of apparatus or the whole office, and make explicit move subsequently. The visualizations will be self-explanatory which can be easily understood by the user. There will be line plots and graphs (choropleth) which can be used as an effective measure while devising any new program.

4.2 **NON-FUNCTIONAL REQUIREMENTS**

Scalability: Analytics platform must be applicable to a machine or facility of any size. The arrangement must most likely include resources without a requirement for any steady interest in equipment, programming or committed work hours.

Performance: The target of a mechanical examination stage is to give a creation office exact and opportune information especially the timely data. Our model will have improved performance because of the use of datasets with lowest time intervals and has high precession. For checking the accuracy we have shown the performance metrics of different Machine learning and Deep learning algorithms and techniques.

Documentation: Coding standards are maintained throughout the project.

Maintainability: This project has easy maintainability, can be modifiable and integrated with advanced computational and operational technologies.

4.3 **HARDWARE REQUIREMENTS**:

System : Core i5 Processor

Hard Disk :1 TB.

Monitor : 15" LED

Input Devices : Keyboard, Mouse

RAM : 8GB.

4.4 **SOFTWARE REQUIREMENTS:**

Operating system: Windows 8.1 or 10 /UBUNTU.

Coding Language : Python

Software : Anaconda -Jupyter Notebook

Data Repository:- USGS's instantaneous values REST service URL generation tool.

Web browser -Google chrome, Firefox, IE8+.

CHAPTER 5

SYSTEM DESIGN

5.1 DATA COLLECTION

This step is essential in light of the fact that the quality and amount of information that we accumulate will legitimately decide how great your predictive model can be. The information for this examination has been gathered from USGS's National Water Information System (NWIS), an online information repository supporting the procurement, handling and long haul storage of water quality over the USA. For our research, we have taken information from 31 water monitoring stations in Georgia. Four parameters have been decided for this study, i.e Temperature, pH, dissolved oxygen (DO) and turbidity. Data from 1 October 2014 to 18 February 2019, with the time interval of 15 minutes has been gotten to complete a successful forecast process using this time series that incorporates date/time, parameters and their measurements along with their measurement units.

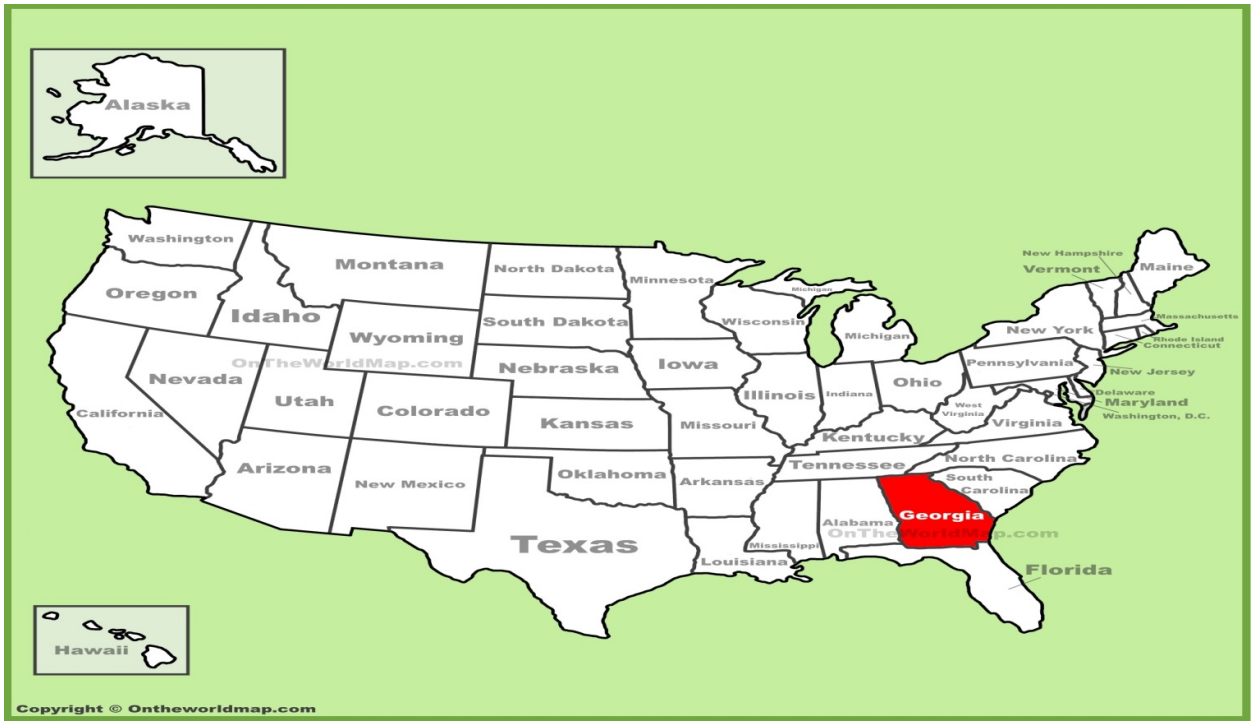


Figure 5.1:- Location of Georgia in the US

Figure 5.2:- County map of Georgia

5.2 **DATA PREPARATION**

We load our information in an appropriate place and set it up to use in our machine learning training. This is likewise a decent time to do any appropriate visualizations of our information, to help check whether there are any pertinent connection between various factors we can exploit, just as to indicate us if there are any imbalances. The dataset gathered from USGS's online information archive initially contained 1,50,729 rows. We have resampled our information by ascertaining every day mean, along these lines by diminishing the dataset with 1603 lines. We will part the dataset into two sections. The initial segment will be most of the dataset, which will be utilized for preparing the model. The training dataset contains 1,339 rows, which is 80% of the total dataset. The second piece of the dataset will be utilized for assessing the model. The testing dataset contains 263 columns, which is 20% of the original dataset. Now and again, the information we gather needs different types of changes and adjustments. Things like de-duping, standardization, error

correction and more are done, all occur in this phase. We have to standardize our information dependent on the model we pick.

	date	temperature	dissolved_oxygen	pH	turbidity
0	2014-10-01	21.350596	6.898147	7.009447	13.468674
1	2014-10-02	21.315316	6.993022	7.044741	10.618085
2	2014-10-03	21.884984	6.716186	6.972112	40.292612
3	2014-10-04	20.344586	6.903656	6.928980	25.303738
4	2014-10-05	17.923966	7.364901	6.980445	26.565599
5	2014-10-06	17.950552	7.529336	7.019258	28.172014
6	2014-10-07	18.914272	7.345588	7.031518	28.425536
7	2014-10-08	20.003922	7.162172	7.031793	27.986241
8	2014-10-09	21.093359	6.888860	7.013097	30.546341
9	2014-10-10	21.421380	6.762587	7.007799	29.287538

Figure 5.3:- Dataset of Georgia

PARAMETERS	MEASUREMENT UNITS
Temperature	℃
pH	No units
Dissolved Oxygen (DO)	mg/L
Turbidity	FNU

Table 5.1:- Parameters with their measurement units

5.3 **CHOOSING A MODEL**

Past examines state that they are a few deficiencies appeared by the conventional ANNs just as the ARIMA models. Thus, to make a powerful water quality prediction model with satisfactory accuracy, we will pick an LSTM display. LSTM has the ability of recalling data for longer timeframes. In this way, The LSTM demonstrate that we have made has a visible layer with 1 input, a hidden layer with 16 LSTM neurons and an output layer that makes a solitary esteem expectation. The ReLU activation function is utilized for the LSTM neurons.

5.4 TRAINING

We will utilize our training dataset to steadily improve our model's capacity to forecast. We can contrast our model's expectations and the output delivered with the output that it should produce and modify the number of neurons in the output layer, to such an extent that we will have progressively accurate forecasts. This process is repeated to train the model. Each cycle of modifying weights and biases is called as a "training step". The LSTM model is trained for 100 epochs and a batch size of 1 is used.

5.5 **MODEL EVALUATION**

This is the place the dataset that we put aside before becomes an integral factor. It enables us to test our model against the information that has never been utilized for preparing. This enables us to perceive how the model may perform against information that it has not yet observed. This is intended to be illustrative of how the model may perform in reality. We will utilize Mean Squared Error (MSE), Root Mean Squared Error (RMSE) and Coefficient of Determination (r2) to assess the execution of our models. We have additionally contrasted the proposed model and the exhibitions of the conventional ANN and ARIMA models.

5.5.1 **Mean Squared Error (MSE)**

MSE calculates the averages of the squares of the error, that is, the average squared difference between the evaluated qualities and what is assessed. It is a risk function, comparing to the estimation of the squared error loss. It evaluates the nature of a predictor (i.e a capacity mapping discretionary contributions to an example of estimations of some arbitrary variable)

In the event that a vector of n forecasts produced from an example of n data pints focuses on all variables and Y is the vector of observed estimations of the factors being anticipated, at that point the within-example MSE of the predictor is figured as:-

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Y'_i)^2 \qquad (5.5.1.1)$$

5.5.2 **Root Mean Squared Error (RMSE)**

The root-mean-square deviation (RMSD) or root-mean-square error (RMSE) is an as often as possible utilized measure of the differences between values (example or populace values) anticipated by a model or an estimator and the values observed. The RMSD speaks to the square root of the second sample moment of the differences between forecasted values and observed values or the quadratic mean of these differences. RMSD is a proportion of accuracy, to contrast between the error of various models for a specific dataset and not between datasets, as it is scale-dependent. RMSD is the square root of the average of squared errors.

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - Y'_i)^2} \tag{5.5.2.1}$$

5.5.3 **Coefficient Of Determination**

The coefficient of determination, indicated by R2 or r2 and pronounced "R squared", is the extent of the variance in the dependent variable that is expected from the independent variable(s).

A data set has n values indicated as $y_1,...,y_n$ (collectively known as y_i or as a vector $y = [y_1,...,y_n]^T$), each associated with a predicted (or modeled) value $f_1,...,f_n$ (known as f_i, or sometimes $\hat{y}_i$, as a vector f).

Define the residuals as $e_i = y_i - f_i$ (forming a vector e).

If $\bar{y}$ is the mean of the observed data:

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{5.5.3.1}$$

Then the change of the dataset can be measured using three sum of squares formulas:

- The total sum of squares (proportional to the variance of the data)

$$SS_{tot} = \sum_i (y_i - \bar{y})^2 \tag{5.5.3.2}$$

- The regression sum of squares, also called the explained sum of squares

$$SS_{reg} = \sum_i (f_i - \bar{y})^2 \quad (5.5.3.3)$$

- The sum of squares of residuals, also called the residual sum of squares

$$SS_{res} = \sum_i (y_i - f_i)^2 = \sum_i {e_i}^2 \quad (5.5.3.4)$$

The most general definition of the coefficient of determination is

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \quad (5.5.3.5)$$

5.6 **PREDICTION OR INFERENCE**

This is the progression where we get the chance to respond to certain inquiries. This is the purpose of this work, where the estimation of machine learning is figured it out. We will anticipate the values one time step ahead and plot the results in like manner.

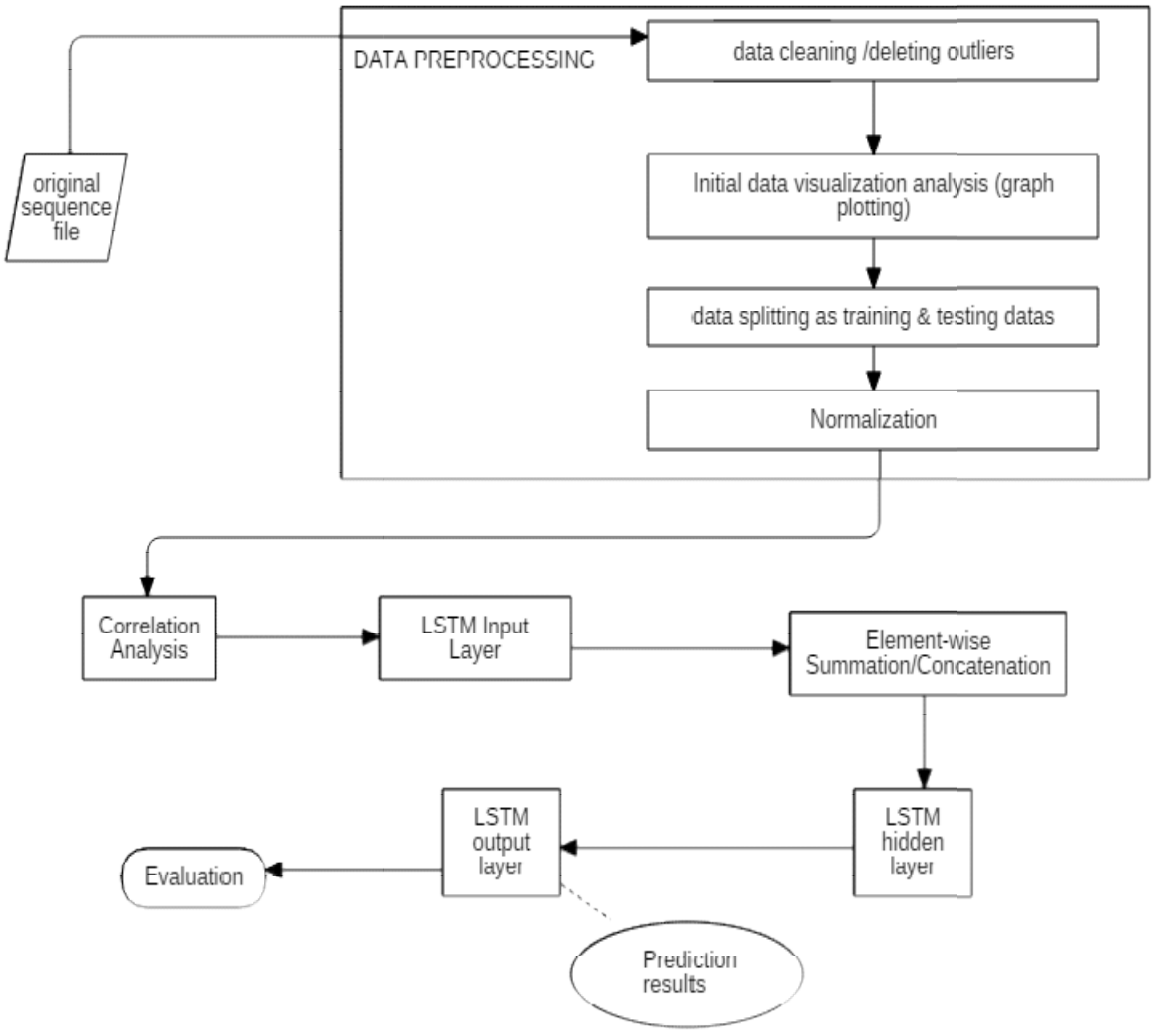


Figure 5.4 :- Architecture diagram of the proposed model

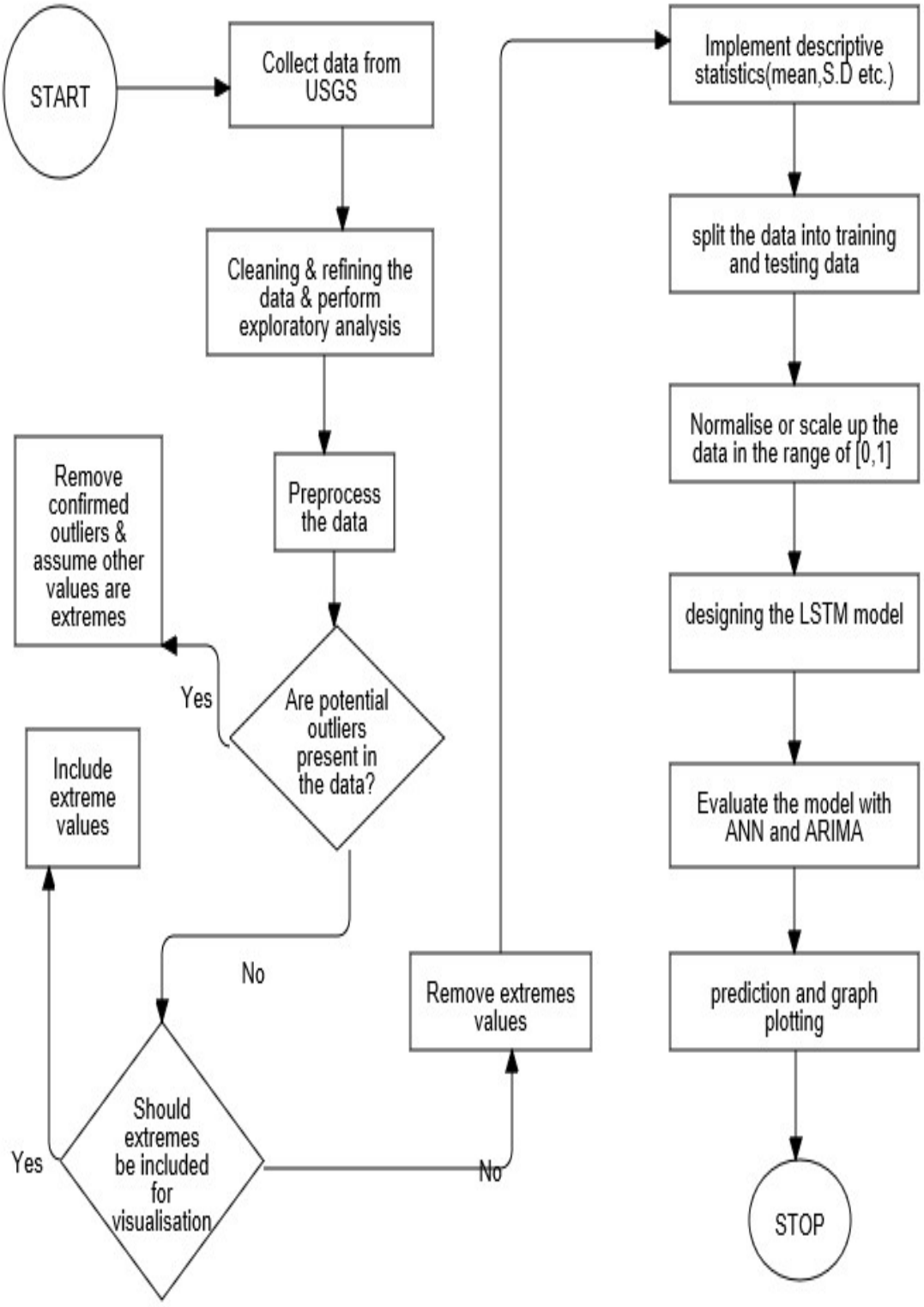


Figure 5.5:- Data flow diagram

5.7 **MODULES AND ITS IMPLEMENTATION**

5.7.1 **Data Cleaning and Preparation**

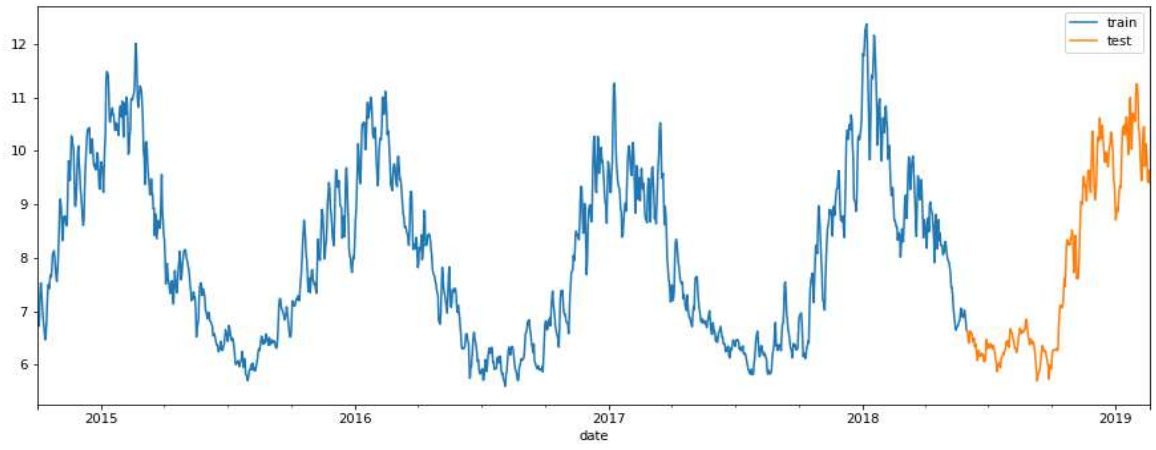
The data that we have collected from the USGS NWIS website, has been converted from a tab-separated values (.tsv) file format to a comma-separated values (.csv) file format. The data will be stored in a dataframe and date column has been set as an index. The rows with NaN values will be dropped from the dataframe. The data has been resampled on a daily basis and the daily average has been calculated. Then, the resampled data will be saved in another CSV file.

The above process will be done for all the collected data from 31 water monitoring stations of Georgia. All the CSV files will be grouped together by the date column and the mean is calculated. The output dataframe will be saved in a CSV file.

The resultant dataset will be used for our model training.

5.7.2 **Data Pre-processing**

The required dataset has been taken and only three columns, i.e. date, dissolved oxygen, and pH, have been used for the computational purposes. Since the values in the date column will be in the string format, we will convert the values in the date-time format and set it as the index. The dataset has been split as training and testing datasets. Training set contains 80%, whereas testing set 20% of the original dataset. MinMax normalization has been used to transform the values of both training and testing datasets in the range of -1 and 1.



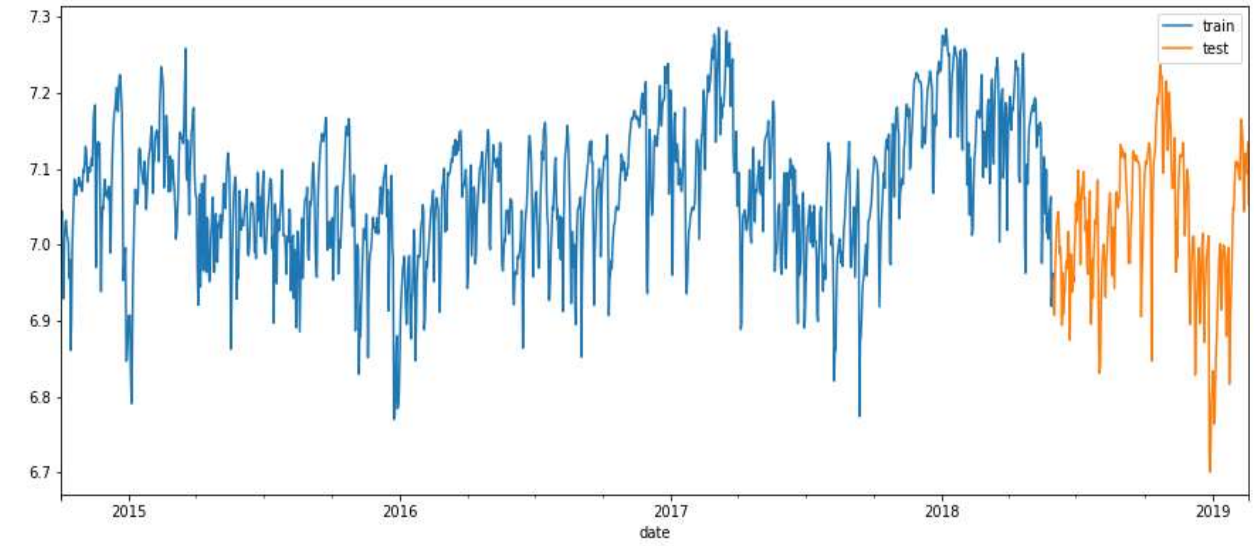


Figure 5.6:- Split of the original dataset

5.7.3 **Creating the model**

To create our LSTM model, first we create a Sequential model, which is a linear stack of layers. We add a visible layer with 1 input, a hidden layer with 16 LSTM neurons, and an output layer that makes a single value prediction. ReLU activation function has been used for the LSTM neurons. ReLu activation function is a piecewise linear function that will yield the input directly, else it will yield 0. This will help in accomplishing better execution, helps in overcoming the vanishing gradient problem, enabling models to adapt quicker and perform better.

5.7.4 **Compiling the model**

Before training the model, the model has to be compiled using the loss function and the optimizer. Mean Squared Error has been used as the loss function and Adam as the optimizer. Adam is an acronym for “Adaptive Moment Estimation”. It is an optimization technique that can be utilized instead of the classical stochastic gradient process to update network weights in a repetitive manner based on our training data.

5.7.5 **Training the model**

The model has been trained with 100 epochs. A batch size of 1 has been used while training. One of the callbacks, EarlyStopping has been used, so that the model will terminate itself when the monitored quantity has finished improving.

5.7.6 **Model Evaluation**

R^2 score, MSE and RMSE between the actual values and the predicted values have been calculated. The values of the metrics has been compared with the traditional ANN model and the ARIMA model. Based on the values, LSTM has shown a better performance than that of others. Hence, LSTM will be chosen for prediction. To achieve more accuracy, the LSTM model can be re-trained.

5.7.7 **Prediction**

Our LSTM model has been used to predict the values. Testing set will be used for the prediction purpose.

5.8 **TOOLS USED**

We have used various tools and modules of python in our model to create our proposed model. We have used tools such as JetBrains PyCharm as well as Jupyter Notebook to work with our project.

- PyCharm is an integrated development environment (IDE) utilized in computer programming, explicitly for the Python language. It gives code analysis, a graphical debugger, an in-built unit tester and supports web development with Django.
- Jupyter Notebook is an open-source application that enables you to make and share reports that contain live code, equations, visualizations, and narrative text. Its uses are data cleaning and modification, numerical simulation, statistical modelling, data visualizations, machine learning and so on.

We have used various libraries of Python to create our water quality prediction model and produce some visualizations.

- **Pandas**:- Pandas is an open-source, BSD-authorized library giving high-performance,easy-to-utilize data structures. It is a software library, utilized for data manipulation and analysis. It gives numerous highlights, for example,

- DataFrame object for data manipulation and coordinated indexing.
- Tools for reading and writing data between in-memory data structures and diverse document formats.
- Data arrangement and incorporated treatment of missing data.
- Label-based slicing, extravagant indexing and subsetting of datasets.
- Data structure column addition and deletion.
- Group by engine permitting split-apply-consolidate activities on datasets.
- Dataset combining and joining
- Various leveled axis indexing to work with high dimensional information in a lower dimensional information structure.
- Time-series functionality:- Date range generation and frequency conversion, moving window measurements, moving window direct regressions, date moving and lagging.
- Gives data filtration.

- **NumPy:**- NumPy is the library, including support for expansive, multi-dimensional arrays and matrices, alongside an accumulation of high-level numerical functions to work on these arrays. Its highlights are:-
  - Python options to MATLAB.
  - n-dimensional arrays.
  - Fourier transforms and shapes control.
  - Linear variable based math and arbitrary number generation.
- **Matplotlib**:- Matplotlib is a Python 2-D plotting library which produces production quality figures in an assortment of hard copy designs and intuitive environments crosswise over platforms. It is utilized for visualizing the information with the assistance of general plots, for example, line plot, histogram, thickness plots, etc.
- **Scikit-learn**:- Scikit-learn is a free ML library for the Python programming language. It highlights different classification, regressions and clustering algorithms including Support Vector Machines, Random Forests, Gradient Boosting, K-means, and DBSCAN. It is intended to interoperate with the Python numerical and logical libraries, NumPy and SciPy. Its highlights are:-
  - diminishing the quantity of arbitrary factors to consider.
  - utilized for differentiating, validating and picking parameters and models.
  - utilized for feature extraction and standardization.

- **Keras**:- Keras is an open-source neural network library. It is equipped for running on TensorFlow. It is intended to empower quick experimentation with deep neural networks, it centers around being easy to use, particular and extensible. It takes into account simple and quick prototyping. It bolsters both convolutional networks and recurrent networks, just as combinations of the two. It runs consistently on CPU and GPU. It designs the model for training.

CHAPTER 6

RESULTS

Below we have plotted the basic line plot for all the four parameters required for predicting water quality.

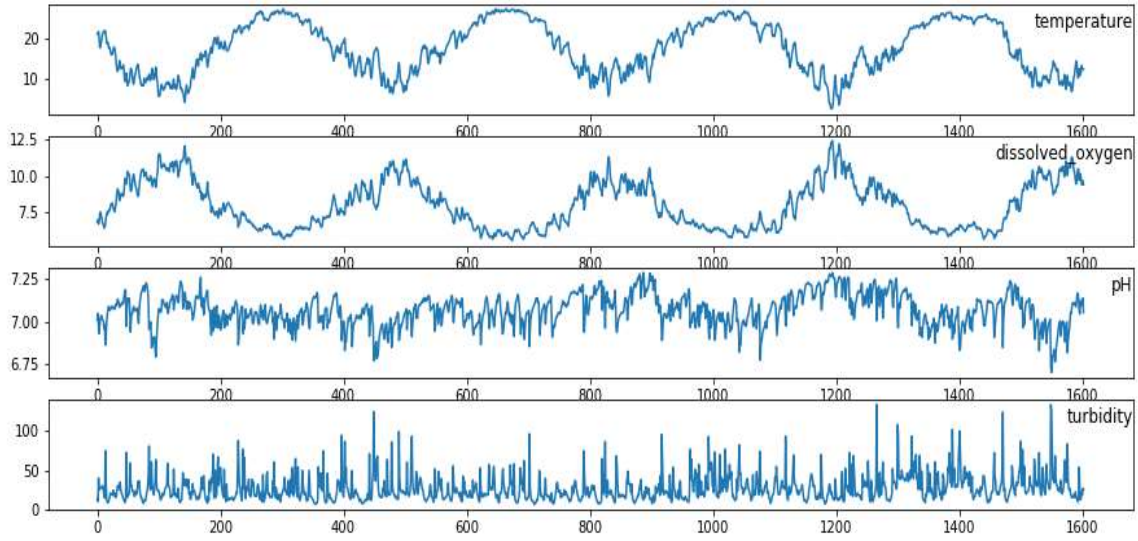


Figure 6.1:- Line plot of four parameters of Georgia dataset

In the line plot of pH, we can see that the values are in the range of 6.7 to 7.3. This shows that the pH of water in Georgia is in the ideal range. (6.5 to 7.5)

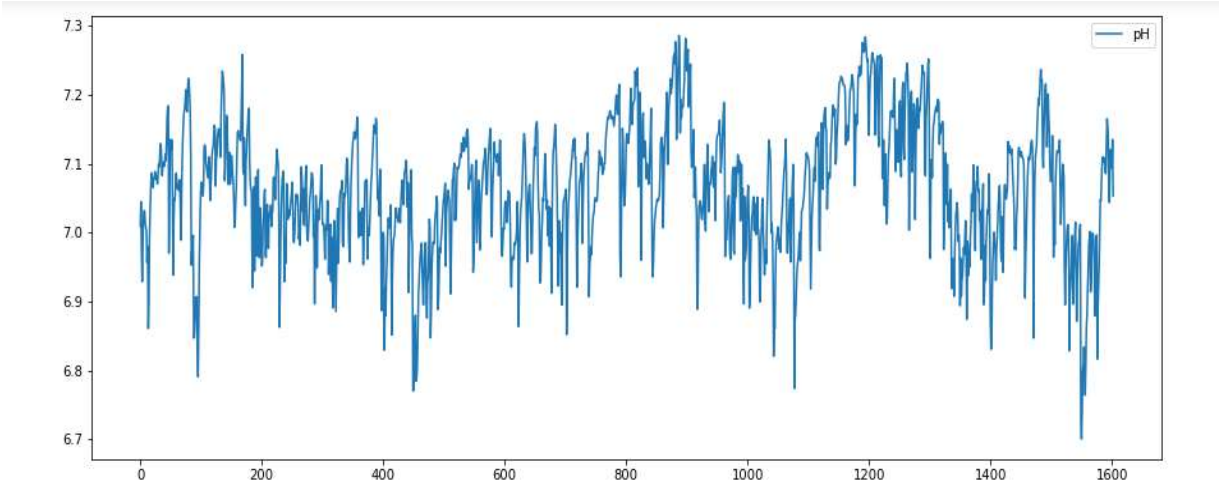


Figure 6.2:- Line plot of pH

We can see from the seasonal decomposition of pH that there is no trend and seasonality is being followed. Since there is no trend or seasonality present, we can say that our data is stationary.

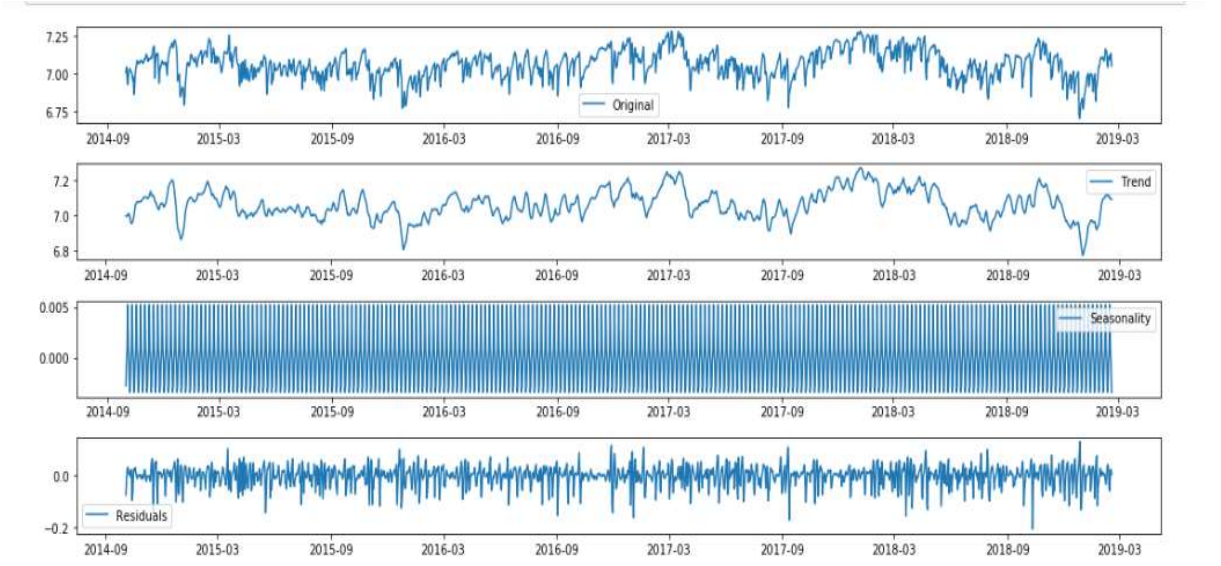


Figure 6.3:- Seasonal Decomposition of the pH

We have trained our model and used it for prediction. Here, we can see that our model has trained in a better manner than the other models.

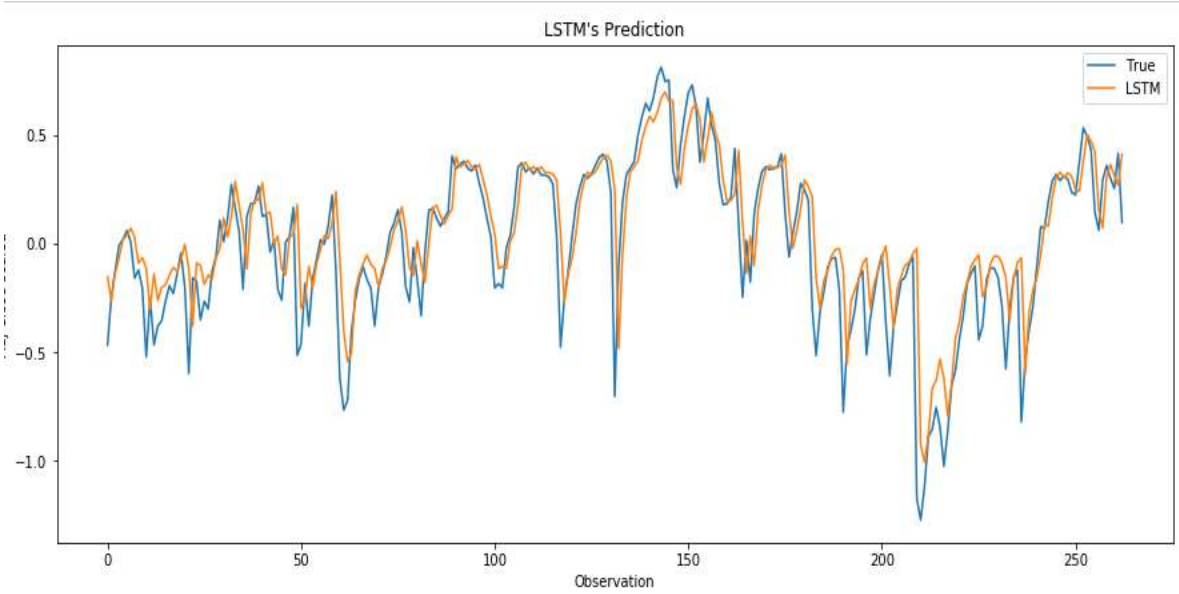


Figure 6.4:- Output of the univariate prediction of pH.

The performances of the three models have been calculated using the three metrics, R^2, MSE and RMSE. Here, we can see that LSTM has performed better than the ANN and ARIMA.

MODEL	R^2	MSE	RMSE
LSTM	0.725	0.0390	0.1974
ANN	0.719	0.0402	0.2004
ARIMA	0.696	0.0512	0.2262

Table 6.1:- Performances of LSTM, ANN and ARIMA

Below is the line plot of the amount of dissolved oxygen present in the waters of Georgia. The amount of DO is in the range of 6.0 to 12.0.

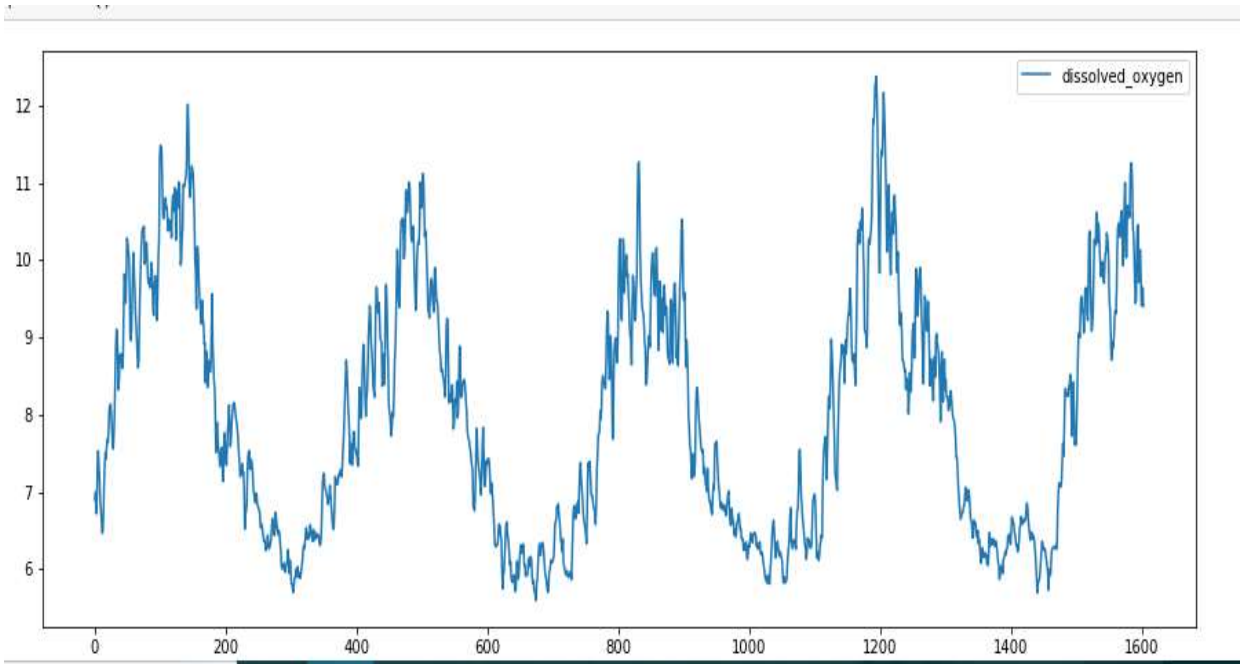


Figure 6.5:- Line plot of DO

We can see from the seasonal decomposition of DO that there is no trend and seasonality is being followed. Since there is no trend or seasonality present, we can say that our data is stationary.

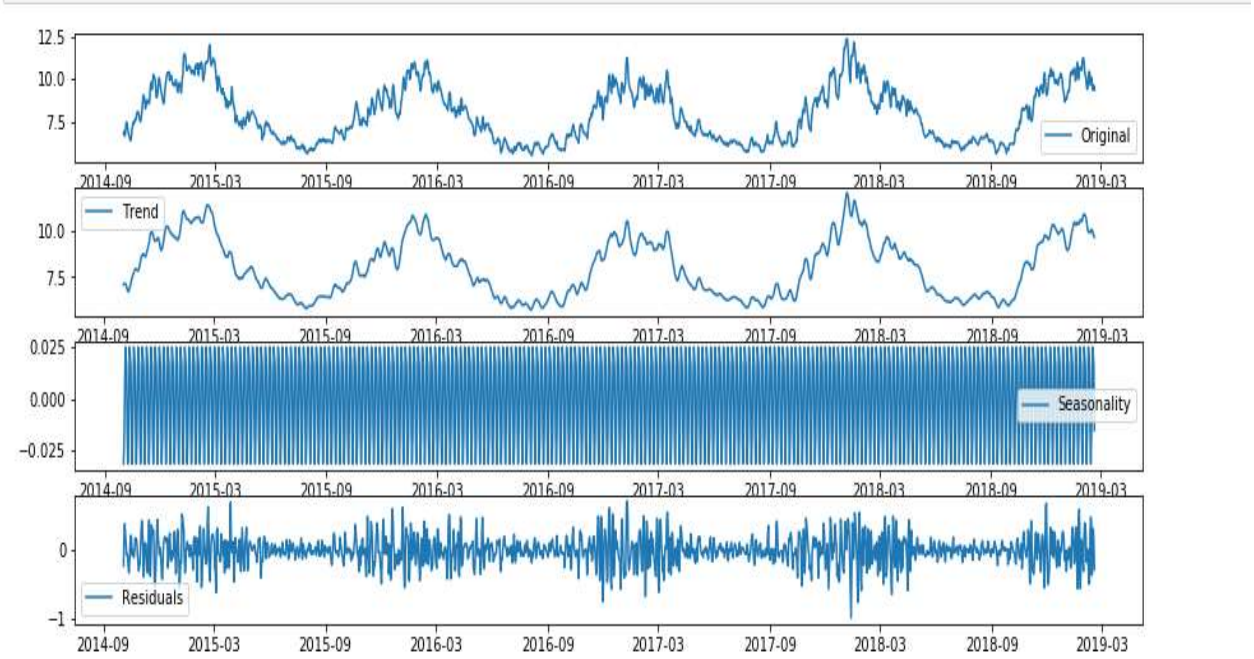


Figure 6.6:- Seasonal Decomposition of DO

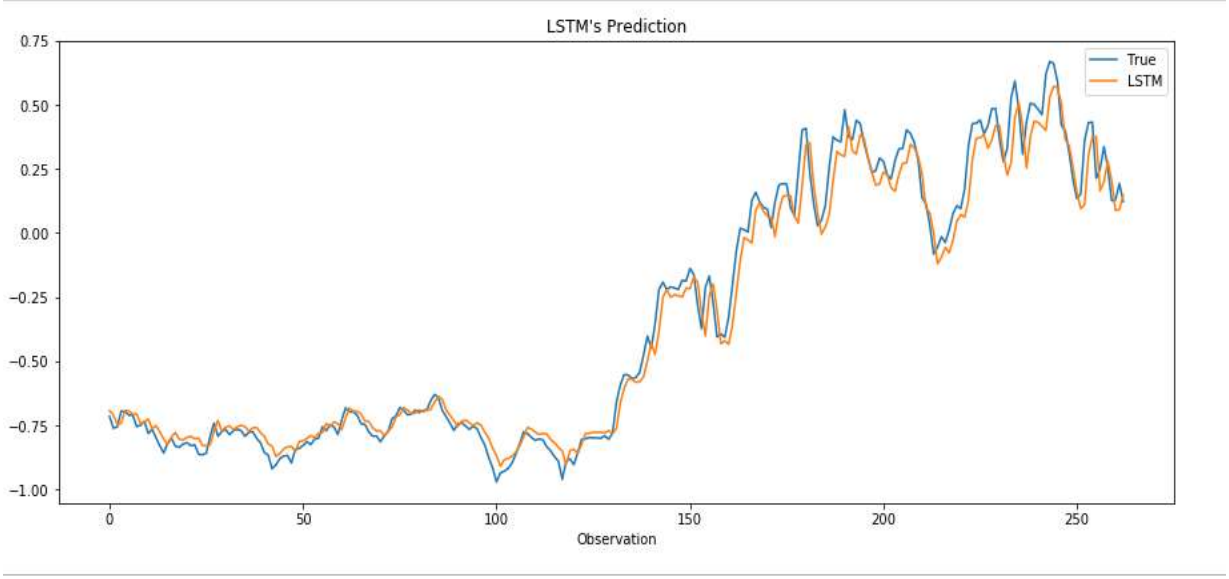


Figure 6.7:- Prediction of DO done by LSTM

The performances of the three models have been calculated using the three metrics, R^2, MSE and RMSE. Here, we can see that LSTM has performed better than the ANN and ARIMA.

MODEL	**R^2**	**MSE**	**RMSE**
LSTM	0.980	0.0060	0.0774
ANN	0.961	0.0101	0.1004
ARIMA	0.843	0.0112	0.1058

Table 6.2:- Performance of LSTM, ANN and ARIMA based on DO

Hence, we can say that LSTM has performed better than the other three models. But this can be further trained to get better results.

CHAPTER 7

TESTING

TESTING OBJECTIVES

The purpose of testing is to discover unwanted errors and bugs. Testing is a process of trying to figure or discover every conceivable faults ,defects or weakness in a working project. It provides different methods to check the functionality of the components, sub-assemblies, assemblies, integration mechanism and security systems etc. of a project.Testing a machine learning model, either LSTM or ANN or ARIMA can be done with the help of two techniques:- one with the dual coding and another by testing the model with different data slices.

7.1 DUAL CODING

With dual coding technique, the idea is to build different models based on different algorithms and comparing the prediction from each of these models given a particular dataset.

MODEL	R^2	MSE	RMSE
LSTM	0.725	0.0390	0.1974
ANN	0.719	0.0402	0.2004

Table 7.2:-Performance of the two models

7.2 TESTING THE DATA WITH DIFFERENT DATA SLICES

We have taken three different data samples with different train and test data splits. The performance of both models, LSTM and ANN has been evaluated based on coefficient of determination and the number of epochs at which the models have stopped training itself.

TEST DATA	LSTM	ANN
263 samples	0.720	0.697
415 samples	0.785	0.771
355 samples	0.743	0.709

Table 7.3:- Performance evaluation based on R2 score for different test samples.

R2 score evaluation table provides an indication of the goodness of fit of a set of predictions to the actual values**.**

TEST DATA	LSTM	ANN
263 samples	28	68
415 samples	24	44
355 samples	29	30

Table 7.4:- Performance evaluation of the models based on no. of epochs on different test samples

CHAPTER 8

CONCLUSION

Water quality prediction has increasingly commonsense importance for the administration of water assets as well as for the counteractive action of water contamination. In light of the successive attributes of water quality markers, this paper proposes another strategy dependent on LSTM NN for water quality forecast and sets up an expectation display based on deep learning. This project proposes new LSTM NN model to predict the value of pH which is the one of the main indicator of water quality in determining the aquatic species habitations with the dumping of industrial wastes and drinking water portability.

The model is prepared by the recorded information of water quality parameters which are provided by the USGS website which monitors the stations and maintains the database and instantaneous values can be generated by USGS REST service URL tool. The datasets contain the water quality parameters information like temperature, DO, pH and turbidity from the year 2014 to 2019(till 18th Feb) for Georgia State with a 15 minute of intervals. To remove the noisiness and inconsistency, the data cleaning and data scaling (normalisation) is carried out along with the graphical presentations and exploratory analysis. To improve the prescient precision of the model, a few re-enactments and parameter determination are completed

.Contrasted with the famous forecasting models ANN and ARIMA, the prescient exactness of LSTM NN is higher which is plotted by graphs and depicted in performance metrics in terms of RMSE, MSE & R^2. In addition, LSTM NN is more speculation. The experiment shows that the RMSE for the LSTM NN is 0.072 which is better than other two famous forecasting models. The prediction results are stable. Considering about the disadvantage of a long preparing cycle or long training cycles, an increasingly successful memory block will be structured in future work.

CHAPTER 9

FUTURE ENHANCEMENT

Our model is as of now fit for univariate parameters for convenient model training. It can, later on, be reached out to cover different parameters whose information is accessible in our dataset. The time series information, with more calculation control, can be done at an hourly premise. This gives a progressively penetrated examination of the time series information. With an appropriate comprehension of the science behind Support Vector Regression, we can actualize it utilizing the Support Vector Machine execution accomplished for this task. We expect to continue examining approaches to improve our model and its forecasts by contemplating changes on the neural system engineering and diverse methodologies for pre-handling the information just as including new unique highlights. In future, we will add more measurements to improve working condition forecasting in a more extravagant condition. We will likewise ponder underlying drivers of deficiencies, and examine its attributes by other time series regression and classification methods, for example, the one utilizing deep belief networks. Considering the disadvantage of long training cycle, a more effective memory block will be designed in the future work. Because of the simplification of the contribution to this model, it has wide-going relevance on a few grouping displaying undertakings, for example, content investigation, music acknowledgement and voice recognition. Besides, because of its little size and productivity, it very well may be effectively sent to continuous frameworks or inserted frameworks. Extra research is to be done on understanding why the Attention LSTM cell is fruitless in coordinating the execution of the general LSTM cell on a portion of the datasets.

CHAPTER 10

REFERENCES

[1] Y. Wang, J. Zhou, K. Chen, Y. Wang & L. Liu. See, "Water Quality Prediction Method Based on LSTM Neural Network," *IEEE 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE) 2017.*

[2] Sun Ruiqi, "A Study on Price Forecasting of US Stock Based on LSTM Neural Network," *Beijing: Capital University of Economics and Business, 2015*.

[3] X. Ma and Z. Tao, "Long Short Term Memory Neural Network for Traffic Speed Prediction Using Remote Microwave Sensor Data," *Transportation Research Part C Emerging Technologies. Vol. 54, May 2015, pp. 187-197*.

[4] W. Deng, G. Wang, X. Zhang, Y. Guo, & G. Li. See, "Water Quality Prediction Based on a Novel Hybrid Model of ARIMA and RBF Neural Network," *IEEE 3rd International Conference on Cloud Computing and Intelligence Systems* 2014.

[5] Joy Parmar & Mosin I Hasan, "Forecasting of River Water Quality Parameters," *International Journal Of Scientific Research in Engineering – IJSRE, May, 2015*.

[6] X. Li and J. Song, "A New ANN-Markov Chain Methodology for Water Quality Prediction," *International Joint Conference on Neural Networks, pp. 12-17 July, 2015*.

[7] S. Jaloree, A. Rajput and S. Gour, " Decision tree approach to build a model for water quality," *Binary Journal of Data Mining & Networking 4 (2014) 25-28*.

[8] M. Ranjbar and M. Khaledian, "Using ARIMA Time Series Model in Forecasting the Trend of Changes in Qualitative Parameters of Sefidrud River," *International Research Journal of Applied and Basic Sciences ,2014 Available online at www.irjabs.com,ISSN 2251-838X / Vol, 8 (3): 346-351 Science Explorer Publications.*

[9] Y. Khan and C. S. See, "Predicting and Analyzing Water Quality using Machine Learning: A Comprehensive Model," *IEEE Long Island Systems, Applications and Technology Conference (LISAT)*, 2016.

[10] H. Liao and W. Sun. "Forecasting and evaluating water quality of Chao Lake based on an improved decision tree method." *Procedia Environmental Sciences* 2 (2010): 970-979.

[11] W. Yao, P. Huang and Z.Jia, "Multidimensional LSTM Networks to predict wind speed", *Proceedings of the 37th Chinese Control Conference*, pp. July 25-27, 2018, Wuhan, China.

[12] X. Shi, Z.Chen, H.Wang, D.Y.Yeung, "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting", *Advances in Neural Information Processing Systems 28* (NIPS 2015).

[13] Liu Fang and Hu Caihong, "Establishment and Application of Forecast Model for Precipitation Based on Mathematical Statistics," *Meteorological and Environmental Sciences, vol. 37, 2014, pp. 89-93, doi: 10.3969/j. ssn. 1673-7148.2014.02.015.*

[14] Sun Yuanhuan and Hu Yuzhuo, "Application of Improved Gray Neural Network Model for Water Quality Prediction," *Chongqing: Chongqing University, 2010*.

[15] Sun Xihao and Yan Lei, "Prediction Model of Taihu Using Improved Fuzzy Time Series," Science and Technology&Innovation, vol. 21, 2016, pp. 15-16, doi: 10.15913/j. Cnki. kjycx.2016.21.015.

[16] Yuan Honglin and Gong Ling, "Using BP Neural Network for The Prediction of Soap River Water Quality," Journal of Security and Environment, vol. 13, Apr. 2013, pp. 106-110, doi: 10. 3969/j. issn. 1009-6094. 2013. 02.023.

[17] http://www.wildml.com/2015/09/recurrent-neural-networks-tutorialpart-1-introduction-to-rnns/

[18] A. Graves, Supervised Sequence Labeling with Recurrent Neural Network, Poland: Studies in Computational Intelligence, July 2012, pp. 37-44.

[19] Hochreiter, S and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, 1997, pp. 1735-1780.

[20] Yoshua Bengio and Patrice Simard." Learing Long-Term Dependencies with Gradient Descent is Difficult," Transactions on Neural Networks , 1994, pp. 157-166.

[21] Bunchingiv Bazartseren and Gerald Hildebrabt, "Short Term Water Level Perdiction Using Neural Networks and Neuro-fuzzy Approach," Neurocomputing, vol. 55, 2003, pp. 439-450.

[22] G.Peter, "Time Series Forecasting Using a Hybrid ARIMA and Neural Network Model," NeuroComputing. Vol. 50, 2003, pp.159-175.

[23] Christopher Olah, "Understanding LSTM Networks", http://colah.github.io/posts/2015-08-Understanding-LSTMs/

[24] Susan Li, "An Introduction on Time Series Forecasting with Simple Neural Networks & LSTM", https://towardsdatascience.com/an-introduction-on-time-series-forecasting-with-simple-neura-networks-lstm-f788390915b

[25] Eugine Kang, "Time Series: ARIMA Model", https://medium.com/@kangeugine/time-series-arima-model-11140bc08c6

[26] Jason Brownlee, “Time Series Prediction with LSTM Recurrent Neural Networks in Python with Keras”, https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-python-keras/

[27] Shi Yan, “Understanding LSTM and its diagrams”, https://medium.com/mlreview/understanding-lstm-and-its-diagrams-37e2f46f1714

[28] Farhad Mallik, “Forecasting Exchange Rates Using ARIMA In Python”, https://towardsdatascience.com/forecasting-exchange-rates-using-arima-in-python-f032f313fc56

APPENDIX A:- SUPPLEMENTARY INFORMATION

1 VANISHING GRADIENT

The vanishing gradient problem is a complication observed in training artificial neural networks with gradient-based learning methods and backpropagation. In some methods, each of the neural network's weights gets an improvement corresponding to the partial derivative of the error function with respect to the present weight in each epoch of training. The difficulty is that in some cases, the gradient will be negligibly small, effectively stopping the weight from updating its value. In the worst case, this may completely prevent the neural network from further training.

Vanishing gradients can be decreased by utilizing the Long Short-Term Memory (LSTM) memory units or using the Rectified Linear Unit (ReLU) activation function, instead of sigmoid and hyperbolic tangent functions.

2 EXPLODING GRADIENT

An error gradient is a direction and magnitude determined amid the training of a neural network that is utilized to improve the network weights in the right way and by the proper amount. In deep networks or recurrent neural networks, error gradients can aggregate during improvement and outcomes to very large gradients. These, in turn, produces large improvements to the network weights, and in turn, an inconsistent network. At a maximum, the values of weights can happen to be so large as to overflow and produce results in NaN values. Exploding gradients can outcomes to an unstable network that at best cannot prepare from the training data and at worst results in NaN weight values that can no longer be improved.

There are some minute indications that tell that you may be suffering from exploding gradients during the training of your network, such as:

- The model is not able to get resistance on your training dataset (e.g. bad loss).
- The model is not stable, producing outcomes in large changes in loss from improvement to improvement.
- The model loss progresses to NaN during training.

Exploding gradients can be decreased by utilizing the Long Short-Term Memory (LSTM) memory units and maybe related gated-type neuron structures.

APPENDIX B: SOURCE CODE

A: DATA PREPARATION

```
import pandas as pd
from datetime import datetime
import numpy as np

df=pd.read_csv(r'H:\Major Project\data of us\georgia\01400500.csv',parse_dates=['datetime'],dayfirst=True)
df['date']=pd.to_datetime(df['datetime'])
df=df.set_index('date')
df.drop(['datetime'],axis=1,inplace=True)
sample=df.resample('D').mean()
sample.to_csv('01400500_modified.csv')
```

```
import pandas as pd
import glob

path="C:/Users/HARI/PycharmProjects/sampleProject/georgia"
allFiles = glob.glob(path+"/*.csv")

list_ = []

for file_ in allFiles:
    df = pd.read_csv(file_,index_col=None, header=0)
    list_.append(df)

frame = pd.concat(list_, axis = 0,sort=True)
keep_col=['date','temperature','dissolved_oxygen','pH']
new_f=frame[keep_col]


fl=new_f.groupby('date').mean()
fl.to_csv('georgia.csv')
```

B: CREATING AND TRAINING THE MODEL

In [59]:
```
import pandas as pd
import numpy as np
%matplotlib inline
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import r2_score
from keras.models import Sequential
from keras.layers import Dense
from keras.callbacks import EarlyStopping
from keras.optimizers import Adam
from keras.layers import LSTM
```

]:
```
scaler = MinMaxScaler(feature_range=(-1, 1))
train_sc = scaler.fit_transform(train)
test_sc = scaler.transform(test)
```

```
C:\Users\HARI\Anaconda3\lib\site-packages\sklearn\preprocessing\data.py:321: DeprecationWarning: Passing 1d arrays as data is d
eprecated in 0.17 and will raise ValueError in 0.19. Reshape your data either using X.reshape(-1, 1) if your data has a single
feature or X.reshape(1, -1) if it contains a single sample.
  warnings.warn(DEPRECATION_MSG_1D, DeprecationWarning)
C:\Users\HARI\Anaconda3\lib\site-packages\sklearn\preprocessing\data.py:356: DeprecationWarning: Passing 1d arrays as data is d
eprecated in 0.17 and will raise ValueError in 0.19. Reshape your data either using X.reshape(-1, 1) if your data has a single
feature or X.reshape(1, -1) if it contains a single sample.
  warnings.warn(DEPRECATION_MSG_1D, DeprecationWarning)
C:\Users\HARI\Anaconda3\lib\site-packages\sklearn\preprocessing\data.py:356: DeprecationWarning: Passing 1d arrays as data is d
eprecated in 0.17 and will raise ValueError in 0.19. Reshape your data either using X.reshape(-1, 1) if your data has a single
feature or X.reshape(1, -1) if it contains a single sample.
  warnings.warn(DEPRECATION_MSG_1D, DeprecationWarning)
```

In [78]:
```
print('Train shape: ', X_train_lmse.shape)
print('Test shape: ', X_test_lmse.shape)
```

```
Train shape:  (1339, 1, 1)
Test shape:  (263, 1, 1)
```

In [79]:
```
lstm_model = Sequential()
lstm_model.add(LSTM(16, input_shape=(1, X_train_lmse.shape[1]), activation='relu', kernel_initializer='lecun_uniform', return_seq
lstm_model.add(Dense(1))
lstm_model.compile(loss='mean_squared_error', optimizer='adam')
early_stop = EarlyStopping(monitor='loss', patience=2, verbose=1)
history_lstm_model = lstm_model.fit(X_train_lmse, y_train, epochs=100, batch_size=1, verbose=1, shuffle=False, callbacks=[early_s
```

```
Epoch 1/100
1339/1339 [==============================] - 5s 3ms/step - loss: 0.0466
Epoch 2/100
1339/1339 [==============================] - 4s 3ms/step - loss: 0.0332
Epoch 3/100
1339/1339 [==============================] - 4s 3ms/step - loss: 0.0325
Epoch 4/100
1339/1339 [==============================] - 3s 3ms/step - loss: 0.0321
Epoch 5/100
1339/1339 [==============================] - 3s 3ms/step - loss: 0.0317
Epoch 6/100
1339/1339 [==============================] - 4s 3ms/step - loss: 0.0314
Epoch 7/100
1339/1339 [==============================] - 4s 3ms/step - loss: 0.0312
Epoch 8/100
1339/1339 [==============================] - 4s 3ms/step - loss: 0.0311
Epoch 9/100
1339/1339 [==============================] - 4s 3ms/step - loss: 0.0310
```

C: TESTING THE MODEL

```
y_train_pred_lstm = lstm_model.predict(X_train_lmse)
print("The R2 score on the Train set is:\t{:0.3f}".format(r2_score(y_train, y_train_pred_lstm)))
print("The R2 score on the Test set is:\t{:0.3f}".format(r2_score(y_test, y_pred_test_lstm)))
```

```
The R2 score on the Train set is:       0.741
The R2 score on the Test set is:        0.720
```

PAPER PUBLICATION

Paper publication has not yet started. Writing part is in progress.