

PREDICTION OF WEATHER FORECASTING USING ML

MAJOR PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of Degree of

MASTER OF COMPUTER APPLICATIONS

From

The University of Kerala, Thiruvananthapuram

By

JOSEY A JOSE

Reg.No:95518455035

Under the guidance of

PRIJI PUNNOOSE

Assistant Professor



**DEPARTMENT OF COMPUTER APPLICATIONS
MAR THOMA INSTITUTE OF INFORMATION TECHNOLOGY
CHADAYAMANGALAM**

2021

PREDICTION OF WEATHER FORECASTING USING ML

MAJOR PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the degree of

MASTER OF COMPUTER APPLICATIONS

From

The University of Kerala, Thiruvananthapuram

By

JOSEY A JOSE

Reg.No:95518455035

Under the guidance of

PRIJI PUNNOOSE

Assistant Professor



**DEPARTMENT OF COMPUTER APPLICATIONS
MAR THOMA INSTITUTE OF INFORMATION TECHNOLOGY
CHADAYAMANGALAM P.O, AYUR
KOLLAM-691534**

2021

MAR THOMA INSTITUTE OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER APPLICATIONS
CHADAYAMANGALAM



CERTIFICATE-I

Certified that this is the Bonafide Record of the Project work entitled

PREDICTION OF WEATHER FORECASTNG USING ML

Submitted By,

JOSEY A JOSE

*In partial fulfillment for the award of **Post Graduate Degree in Computer Applications**
of the **University of Kerala**, carried out at **SCIPY innovative solutions Pvt.Ltd**,
Thiruvananthapuram during the **Sixth Semester** academic under our supervision.*

Internal Guide

Asst.Prof. Priji Punnoose

Head of the Department

Asso.Prof. Priji Kurian Isac

Principal

Prof. Dr. K. Jacob

MAR THOMA INSTITUTE OF INFORMATION TECHNOLOGY

DEPARTMENT OF COMPUTER APPLICATIONS

CHADAYAMANGALAM



CERTIFICATE-II

*Certified that this is the Bonafide Record of the Project work done by the student in the partial fulfillment for the award of the **Post Graduate Degree in Computer Applications** from the **University of Kerala, Thiruvananthapuram.***

Name of the Candidate : JOSEY A JOSE

Reg No : 95518455035

Place : Ayur

Date :

Internal Guide

Asst.Prof. Priji Punnoose

Head of the Department

Asso.Prof. Priji Kurian Isac

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

Let me have the opportunity to thank all those who have been directly or indirectly involved in making my project is a success. First of all, I am grateful to GOD Almighty, for helping me to select this project and giving me the hunger and interest to pursue this interesting.

I am thankful to **Prof. Dr. K. Jacob, Principal, Mar Thoma Institute of Information Technology, Ayur**, for his legal support and permission to do the project.

I am indebted to **Asso. Prof. Priji Kurian Isac, Head of the Computer Applications Department, Mar Thoma Institute of Information Technology, Ayur**, for his support and help in presenting my project.

I express my sincere thanks to **Asst. Prof. Priji Punnoose, Department of Computer Applications, Mar Thoma Institute of Information Technology, Ayur**, for her guidance and assistance in my project work.

I am immensely grateful to **Mr. Rahul Bose CP, SCIPY innovative solution, Thiruvananthapuram**, where I have done my project, for his guidance, encouragement and support during the course of this project.

Last but not the least; I express my gratitude to my parents and friends who have given me inspirations, mental supports and lots of help and encouragement for during this project successfully.

JOSEY A JOSE

ABSTRACT

Weather Prediction has been one of the rapidly emerging technology for ages. The Weather Prediction is a method to forecast the situations and climatic conditions by using various attributes like previous data, scientific knowledge, and experimental observations on the weather over a long time. There have been different weather prediction models evolving for ages. Each time a new model that overcomes the disadvantages of the existing model has been built. The objective of building a new model is to create a perfect and cost-efficient weather prediction model to preserve the growing expenses. This various techniques and models that have been used for Weather prediction. Here this system will predict weather based on parameters such as temperature, humidity and wind. System will take this parameter and will predict weather from previous data in dataset. The system will calculate weather based on these data. Weather forecasting system takes parameters such as temperature, humidity, and wind and will forecast weather based on previous record therefore this prediction will prove reliable. This system can be used in Air Traffic, Marine, Agriculture, Forestry, Military, and Navy etc. In proposed system we are using Machine learning algorithms ,which plays a major role in predictive analytics, which uses the current and past historical data sets to discover knowledge from it and by using that data it the predict future occurrences.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ACKNOWLEDGEMENT	I
	ABSTRACT	II
	LIST OF TABLE	VI
	LIST OF ABBREVIATIONS	VII
1	INTRODUCTION	1
	1.1 ABOUT THE PROJECT	1
2	SYSTEM REQUIREMENTS	2
	2.1 HARDWARE REQUIREMENTS	2
	2.2 SOFTWARE REQUIREMENTS	2
	2.3 TECHNOLOGIES USED	3
	2.3.1 Python	3
	2.3.2 Django	4
	2.3.3 NumPy	4
	2.3.4 Pandas	5
	2.3.5 Scikit-Learn	5
	2.3.6 Machine Learning	5
3	LITERATURE REVIEW	8
4	PROBLEM DEFINITION	9
	4.1 INTRODUCTION	9
	4.2 EXISTING SYSTEM	9
	4.2.1 Limitations of existing system	9

4.3	PROPOSED SYSTEM	10
4.3.1	Merits of proposed system	10
4.4	FEASIBILITY STUDY	10
4.4.1	Technical feasibility	11
4.4.2	Operational feasibility	11
4.4.3	Economic feasibility	11
5	SYSTEM DESIGN	12
5.1	INTRODUCTION	12
5.2	ADOPTION MODULES	12
5.2.1	Data Mining &Preprocessing	12
5.2.2	ML Model Building from Scikit-Learn	13
5.2.3	Model Training & Correlation Analysis	13
5.2.4	Model Testing and Result Analysis	14
5.3	DATA FLOW DIAGRAM	15
5.3.1	Advantages	15
5.3.2	Rules for constructing a data flow diagram	15
5.3.3	Components of data flow diagram	15
5.3.4	Context level diagram	18
5.4.5	Level 1 DFD	18
5.4.6	Level 2 DFD	19
5.4.7	Level 3 DFD	19
5.4	UML DIAGRAM	20
5.4.1	Use case diagram	20
5.4.2	Modular diagram	22
5.4.3	Sequence diagram	23

6	SYSTEM IMPLEMENTATION	24
6.1	INTRODUCTION	24
6.2	ALGORITHMS	25
	6.2.1 Linear Regression Algorithm	25
	6.2.2 Decision Tree Algorithm	25
7	SYSTEM TESTING	26
7.1	INTRODUCTION	26
7.2	LEVELS OF TESTING	27
	7.2.1 Unit testing	27
	7.2.2 Integration testing	27
	7.2.3 System testing	28
	7.2.4 Acceptance testing	28
7.3	TESTING TECHNIQUES	29
	7.3.1 BLACK-BOX TESTING	29
	7.3.2 WHITE-BOX TESTING	29
	7.3.3 GREY-BOX TESTING	29
7.4	TEST CASE AND OUTPUT	30
8	CONCLUSION AND FUTURE ENHANCEMENT	32
8.1	CONCLUSION	32
8.2	FUTURE ENHANCEMENT	32
	APPENDICES	33
	APPENDIX A	33
	REFERENCES	37

LIST OF TABLES

TABLE NO.	TABLE NAME	PAGE NO.
7.4.1	TEST CASE AND OUTPUT	30

LIST OF ABBREVIATIONS

DFD	-	Data Flow Diagram
UML	-	Unified Modeling Language

CHAPTER 1

INTRODUCTION

1.1 ABOUT THE PROJECT

The Weather prediction system has very important in our daily life. It is employed in various fields like agriculture, industries, sports, etc. It is the prediction of weather depending on the previous datasets. The weather prediction seeks a lot of attention nowadays because of its importance. Specific pre-processing techniques like data cleaning, data integration, data reduction, and Data transformation. Data cleaning includes filling up the missing values. It is done either manually or with global constants or by the average of the remaining costs. Some systems follow the method of filling the missing values by replacing the missing values with the average value of the class attribute. In some cases, the missing values are replaced by the most repeated value, i.e., Mode. After handling the missing data's missing values, the data is cleaned by removing the unwanted values, which is also called the Noise. The most effective way to handle Noisy data is by Binning. The data is binned either by Means or Medians or by boundaries. Some other systems even follow the regression or Outlier Analysis to handle the Noise present in the data. Data integration is merging the data or collection of data at a single side collected from multiple repositories. Though there are certain issues like Tuple Duplication, Data redundancy, and Entity identification problems that arise in the data integration, the issues can be handled easily by various techniques like Correlation analysis. After the Pre-processing of the data, the model is built. The model is then trained by a training data set that is specifically used to train the data. After the model is trained, the model is applied with a classifier that classifies the data from a user-specified Test dataset. Thus, the classification of data is done. The above pre-processing and classification are common to all the data mining models. It varies on the train set and test set given to the model. In the proposed the model is presented with a weather dataset as the aim is Weather Prediction.

CHAPTER 2

SYSTEM REQUIREMENTS

2.1 HARDWARE REQUIREMENTS

Processor	:Intel P4 and above
Processor speed	:1.7 GHZ
Random Access Memory	:1 GB
Hard Disk Memory	:240 GB
Monitor	:Colour Monitor

2.2 SOFTWARE REQUIREMENTS

Operating System	: Windows /Linux
Front End	: Django
Back End	: Machine Learning

2.3 TECHNOLOGIES USED

A set of programs associated with the operation of a computer is called software. Software is the part of the computer system which enables the user to interact with several physical hardware devices. The minimum software requirement specifications for developing this project are as follows:

Operating System: Windows 10

Documentation Tool: MS Word

2.3.1 PYTHON

Python is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum and first released in 1991. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++ or Java. The language provides constructs intended to enable writing clear programs on both a small and large scale.

Python features a dynamic type system and automatic memory management and supports multiple including object-oriented, imperative, functional programming, and procedural styles. It has a large and comprehensive standard library.

Python interpreters are available for many operating systems, allowing Python code to run on a wide variety of systems. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython is managed by the non-profit Python Software Foundation(PSF).

Python's development is conducted largely through the Python Enhancement Proposal(PEP) process. The PEP process is the primary mechanism for proposing major new features, for collecting community-input on an issue, and for documenting the design decisions that have gone into Python. Outstanding PEPs are reviewed and commented

Age Estimation and Gender Recognition System up on by the Python community and by Van Rossum, the Python project's Benevolent Dictator For Life.

2.3.2 Django

Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support. It is complete, versatile, secure, scalable, Maintainable and portable. Django follows the "Batteries included" philosophy and provides almost everything developers might want to do "out of the box". Django can be (and has been) used to build almost any type of website — from content management systems and wikis, through to social networks and news sites. It can work with any client-side framework, and can deliver content in almost any format (including HTML, RSS feeds, JSON, XML, etc). Django helps developers avoid many common security mistakes by providing a framework that has been engineered to "do the right things" to protect the website automatically. Django uses a component-based architecture (each part of the architecture is independent of the others, and can hence be replaced or changed if needed). Django code is written using design principles and patterns that encourage the creation of maintainable and reusable code. Django is written in Python, which runs on many platforms. That means that you are not tied to any particular server platform, and can run your applications on many flavors of Linux, Windows, and Mac OS X.

2.3.3 NumPy

NumPy, which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed. NumPy is a Python package. It stands for 'Numerical Python'. It is a library consisting of multidimensional array objects and a collection of routines for processing of array. Numeric, the ancestor of NumPy, was developed by Jim Hugunin. Another package Numarray was also developed, having some additional functionalities. In 2005, Travis Oliphant created NumPy package by incorporating the features of Num array into Numeric package. There are many contributors to this open-source project.

2.3.4 Pandas

pandas is a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labeled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real-world data analysis in Python. Additionally, it has the broader goal of becoming the most powerful and flexible open source data analysis/manipulation tool available in any language. It is already well on its way toward this goal. pandas is well suited for many different kinds of data: Tabular data with heterogeneously typed columns, as in an SQL table or Excel spreadsheet Ordered and unordered (not necessarily fixed-frequency) time series data. Arbitrary matrix data (homogeneously typed or heterogeneous) with row and column labels Any other form of observational / statistical data sets. The data need not be labeled at all to be placed into a pandas data structure.

2.3.5 Scikit-learn

Scikit-learn is machine learning library. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy. Scikit-learn is a Num FOCUS fiscally sponsored project.

2.3.6 Machine Learning

Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values. Labeled data has both the input and output parameters in a completely machine-readable pattern, but requires a lot of human labor to label the data, to begin with. Unlabeled data only has one or none of the parameters in a machine-readable form. This negates the need for human labor but requires more complex solutions.

There are also some types of machine learning algorithms that are used in very specific use-cases, but three main methods are used today.

Supervised Learning - Supervised learning is one of the most basic types of machine learning. In this type, the machine learning algorithm is trained on labeled data. Even though the data needs to be labeled accurately for this method to work, supervised learning is extremely powerful when used in the right circumstances.

In supervised learning, the ML algorithm is given a small training dataset to work with. This training dataset is a smaller part of the bigger dataset and serves to give the algorithm a basic idea of the problem, solution, and data points to be dealt with. The training dataset is also very similar to the final dataset in its characteristics and provides the algorithm with the labeled parameters required for the problem.

The algorithm then finds relationships between the parameters given, essentially establishing a cause and effect relationship between the variables in the dataset. At the end of the training, the algorithm has an idea of how the data works and the relationship between the input and the output.

This solution is then deployed for use with the final dataset, which it learns from in the same way as the training dataset. This means that supervised machine learning algorithms will continue to improve even after being deployed, discovering new patterns and relationships as it trains itself on new data.

Unsupervised Learning - Unsupervised machine learning holds the advantage of being able to work with unlabeled data. This means that human labor is not required to make the dataset machine-readable, allowing much larger datasets to be worked on by the program.

In supervised learning, the labels allow the algorithm to find the exact nature of the relationship between any two data points. However, unsupervised learning does not have labels to work off of, resulting in the creation of hidden structures. Relationships between data points are perceived by the algorithm in an abstract manner, with no input required from human beings.

The creation of these hidden structures is what makes unsupervised learning algorithms versatile. Instead of a defined and set problem statement, unsupervised learning algorithms can adapt to the data by dynamically changing hidden structures

Reinforcement learning - Reinforcement learning directly takes inspiration from how human beings learn from data in their lives. It features an algorithm that improves upon itself and learns from new situations using a trial-and-error method. Favorable outputs are encouraged or 'reinforced', and non-favorable outputs are discouraged or 'punished'. Based on the psychological concept of conditioning, reinforcement learning works by putting the algorithm in a work environment with an interpreter and a reward system. In every iteration of the algorithm,

the output result is given to the interpreter, which decides whether the outcome is favorable or not. In every iteration of the algorithm, the output result is given to the interpreter, which decides whether the outcome is favorable or not.

In case of the program finding the correct solution, the interpreter reinforces the solution by providing a reward to the algorithm. If the outcome is not favorable, the algorithm is forced to reiterate until it finds a better result. In most cases, the reward system is directly tied to the effectiveness of the result.

In typical reinforcement learning use-cases, such as finding the shortest route between two points on a map, the solution is not an absolute value. Instead, it takes on a score of effectiveness, expressed in a percentage value. The higher this percentage value is, the more reward is given to the algorithm. Thus, the program is trained to give the best possible solution for the best possible reward.

CHAPTER 3

LITERATURE REVIEW

Weather forecasting has traditionally been done by physical models of the atmosphere, which are unstable to perturbations, and thus are inaccurate for large periods of time. Since machine learning techniques are more robust to perturbations, in this we explore their application to weather forecasting to potentially generate more accurate weather forecasts for large periods of time.

Nitin Singh et al. proposed a technique of using machine learning algorithms to anticipate the weather. The implemented system use machine learning algorithms and data analytics like the Random forest algorithm to predict the weather. In the proposed system, the application that gets the real-time data from humidity-temperature and pressure from the sensors to predict the possibility of rain has been developed on Raspberry Pi 3B. The GPIO pins of the Raspberry Pi 3 B works between the sensors and the application. shows the overall idea of using Machine learning algorithms in Weather prediction . The application takes and collects the data through the GPIO pins. The model is being trained in accordance with the Random Forest Algorithm. The result is generated by the system in the form of Binary digits (1/0). The result 1 indicates that there would be rainfall, and 0 indicates that there are no chances for rainfall. The resultant confusion matrix has generated 1491 correct predictions that there would be no rainfall and 122 correct predictions that there will be rainfall, which indicates that 1613 correct anticipations have been brought out of 1835 test data.

Afan Galih Salman et al. has come up with a technique of utilizing Deep Learning techniques for weather anticipation and prediction. Various prediction models and their prediction performances have been discussed in this paper. Those forecast models incorporate Recurrence Neural System (RNN), Contingent Confined Boltzmann Machine (CRBM), and Convolutional System (CN) models. The investigation inferred that the Contingent Confined Boltzmann Machine (CRBM) and Convolutional Neural System (CNN) could be used to predict the weather on a concourse of time-series troubles and the Recurrence Neural Network (RNN) which is implemented using heuristically optimization method can be used to predict the rainfall with fair rainfall level.

CHAPTER 4

PROBLEM DEFINITION

4.1 INTRODUCTION

The Weather prediction system has very important in our daily life. It is employed in various fields like agriculture, industries, sports, etc. It is the prediction of weather depending on the previous datasets. The weather prediction seeks a lot of attention nowadays because of its importance. The prediction of weather is not easy. After considering the dataset, it has to undergo specific pre-processing technique. After the Pre-processing of the data, the model is built. The model is then trained by a training data set that is specifically used to train the data. Thus, the classification of data is done. It varies on the train set and test set given to the model. In the proposed the model is presented with a weather dataset as the aim is Weather Prediction.

4.2 EXISTING SYSTEM

The traditional forecast process employed by most NMHSs involves forecasters producing text-based, sensible, weather-element forecast products (e.g. maximum/minimum temperature, cloud cover) using numerical weather prediction (NWP) output as guidance.

As computer technology and high-speed dissemination systems evolved (e.g. Internet), National Weather Service (NWS) customers/partners were demanding detailed forecasts in gridded, digital and graphic formats. Traditional NWS text forecast products limit the amount of additional information that can be conveyed to the user community. The concept of digital database forecasting provides the capability to meet customer/partner demands for more accurate, detailed hydrometeorological forecasts. Digital database forecasting also offers one of the most exciting opportunities to integrate PWS forecast dissemination and service delivery, which most effectively serves the user community.

4.2.1 LIMITATIONS OF EXISTING SYSTEM.

- Time consuming
- Only small area's weather can be predicted.
- Record maintenance issues

4.3 PROPOSED SYSTEM

In proposed system we are using Machine learning algorithms ,which plays a major role in predictive analytics, which uses the current and past historical data sets to discover knowledge from it and by using that data it the predict future occurrences. In this project there are two algorithm such as linear and decision tree for weather forecasting and prediction. we have concluded that linear regression and decision tree is best when predicting weather forecast which have dependent dataset because already we have linear data for linear regression model and decision tree Regression can analyse and decide the results and there by enhance prediction efficiency

4.3.1 MERITS OF PROPOSED SYSTEM

- High accuracy.
- Wide area's weather can be Predict.
- Less Time consuming.

4.4 FEASIBILITY STUDY

The main objective of the feasibility study is to test the economical, technical and operational feasibility while developing the system. This analysis is done by investigating the existing system in the area under generating an idea about the new system. Feasibility study is a test of proposed system regarding its workability, impact on the organization, ability to meet the needs and effective use of resources. Thus, when a new project is proposed, it normally goes through a feasibility study before it is approved for development. The study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that is spend on it.

Three key considerations involved in the feasibility analysis are:

4.4.1 Technical Feasibility

4.4.2 Operational Feasibility

4.4.3 Economic Feasibility

4.4.1 TECHNICAL FEASIBILITY

The site must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the site requirement in the terms of input, output, programs and procedures. Technical feasibility centers around computer system and to what extent it can support the proposed addition. For example, if the current computer system is operating at 80% capacity then running another application could overload the system or requires additional hardware. This involves the financial considerations to accommodate the additional technical enhancements. If budget is not a serious constraint, then the project is judged technically feasible. Since no further addition of hardware or software is needed, the proposed system Reading aid for the blind people using Tesseract and AlexNet.

4.4.2 OPERATIONAL FEASIBILITY

The operational feasibility depends up on whether system performed in the expected way or not. The application developed is so simple and user friendly there is no special user training is required. So this application can be said to be operationally feasible. The proposed system is very much user friendly and operations on it can be done very easily. The language used in is English and every people can operate it reading options. Quick responses for the queries are available. The website is focused on providing a user-friendly environment so that any users can do what they like. The processes like registration, login all are simple processes. There is no need of special training required for the users.

4.4.3 ECONOMIC FEASIBILITY

The developing system must be justified cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. This deals with whether expected cost saving, increase the profits and reductions in required investment, and other benefits exceed the cost of developing and operating the proposed system. Its preliminary investigation is concentrated on costs of hardware and software. The online application entitled Reading aid for the blind people using Tesseract and AlexNet is economically feasible because it reduces the network traffic and the need for high band width mediums.

CHAPTER 5

SYSTEM DESIGN

5.1 INTRODUCTION

System Design develops the architectural details required to build system or product. The system design process encompasses the following activities:

- Partition the analysis model into subsystems
- Identify concurrency that is dictated by the problem.
- Develop design for the user interface.
- Choose a basic strategy or implementing data management.
- Identify global resources and the control mechanisms required to access them
- Design an appropriate control mechanism for the system, including task management.

5.2 ADOPTION OF MODULES

The application consist of Four Modules

- Data mining and preprocessing
- Machine Learning model building from Scikit-learn
- Model training and core relation analysis
- Model testing and result analysis

5.2.1 Data Mining and Preprocessing

Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format. Steps Involved in Data Preprocessing:

1. Data Cleaning:

The data can have many irrelevant and missing parts. To handle this part, data cleaning is done. It involves handling of missing data, noisy data etc.

(a). Missing Data:

This situation arises when some data is missing in the data. It can be handled in various ways.

Some of them are:

Ignore the tuples:

This approach is suitable only when the dataset we have is quite large and multiple values are missing within a tuple.

Fill the Missing values:

There are various ways to do this task. You can choose to fill the missing values manually, by attribute mean or the most probable value.

(b). Noisy Data:

Noisy data is a meaningless data that can't be interpreted by machines. It can be generated due to faulty data collection, data entry errors etc. It can be handled in following ways :

Binning Method:

This method works on sorted data in order to smooth it. The whole data is divided into segments of equal size and then various methods are performed to complete the task. Each segmented is handled separately. One can replace all data in a segment by its mean or boundary values can be used to complete the task.

Regression:

Here data can be made smooth by fitting it to a regression function. The regression used may be linear (having one independent variable) or multiple (having multiple independent variables).

5.2.2 Machine Learning Model Building from Scikit learn

Scikit-learn is an open source Python library that is used to build machine learning model using various algorithms. There are several models that you can choose according to the objective that you might have you will use algorithms of classification, prediction, linear regression.

5.2.3 Model Training and Correlation Analysis

Model training is done by training the datasets to run smoothly and see an incremental improvement in the prediction rate. Remember to initialize the weights of your model randomly - the weights are the values that multiply or affect the relationships between the inputs and outputs- which will be automatically adjusted by the selected algorithm the more you train them. correlation analysis is used to find the correlation between data points on data set there by improve prediction accuracy .

5.2.4 Model Testing and Result Analysis

Model testing is the process of testing the learned properties. The model does not know and verify the precision of your already trained model. so, we use testing data set having 20% data for testing. after the testing we can conclude the model accuracy and if the accuracy is low we can retrain the model either minimize error or choose another algorithm for training .

5.3 DATA FLOW DIAGRAM

A dataflow diagram is a graphical technique that depicts information and transforms that are applied as data move from input to output. The DFD is used to represent increasing information flow and functional details. A level-0 DFD is also called a fundamental system model represents the entire software elements as a single bible with input and output indicated by incoming and outgoing arrows respectively.

Additional process and information flow parts are represented in the next level, i.e., level 1 DFD. Each of the process represented at level 1 are sub functions of overall system depicted in the context model. Any processes that are complex in level 1 level will be further represented into sub functions in the next level, i.e., level 2

5.3.1 ADVANTAGES

- Users easily understood these simple notations.
- Users can make suggestions for modifications.
- They can also spot problem quickly
- If analyst wants to overview the overall system late, they use the higher overview

5.3.2 RULES FOR CONSTRUCTING A DATA FLOW DIAGRAM

- Arrows should not cross each other.
- Squares, circles and files must bear names.
- Decomposed data flow squares and circles can have same names.
- Choose meaningful names for data flow.
- Draw all data flows around the outside of the diagram.

5.3.3 COMPONENTS OF DATA FLOW DIAGRAM

Data Flow Diagram (DFD) is an important tool used by system analyst. DFD provide an overview of what data a system would process, what transformation of data are done, what files are used and where the results flow. The graphical representation of the system makes it a good communication tool between the user and the analyst. Analysis model help us to understand the relationship between different components in the design.

The analysis modeling must achieve three primary objectives.

- To establish a basis for creation of software design.
- To describe what the user requires.
- To define set of requirements that can be validated once the software is built.

A data flow diagram is a graphical technique that depicts information flow and transforms that are applied as data move from input to output. The DFD is used to represent increasing information flow and functional details. A level 0 DFD also called fundamental system model represents the entire software elements as single bubble with input and output indicated by incoming and outgoing arrow respectively. To construct the data flow diagram we use arrows, circle, and rectangles. A Data Flow Diagram (DFD) is a graphical representation of the “flow” of data through an information system, modeling its process aspects.

Components of Data Flow Diagram

There are only four symbols that are used in the drawing of data flow diagrams.

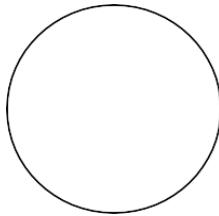
These are explained below together with the rules that apply to them.

External Entities



External entities represent the sources of data that enter the system or the recipients of data that leave the system.

Process



Processes represent activities in which data is manipulated by being stored or retrieved or transformed in some way.

Data Stores



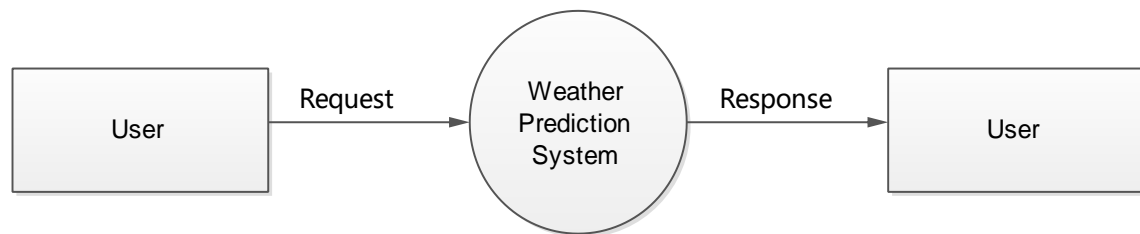
Data stores represent stores of data within the system.

Data Flow

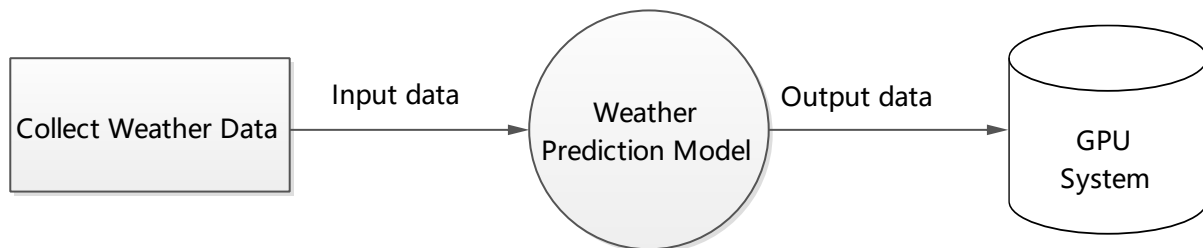


A data flow shows the flow of information from its source to its destination. A line represents a data flow, with arrowheads showing the direction of flow.

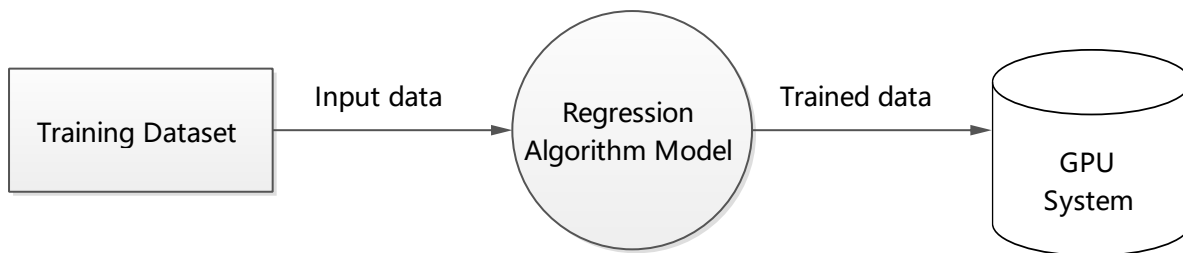
5.3.4 CONTEXT LEVEL DIAGRAM



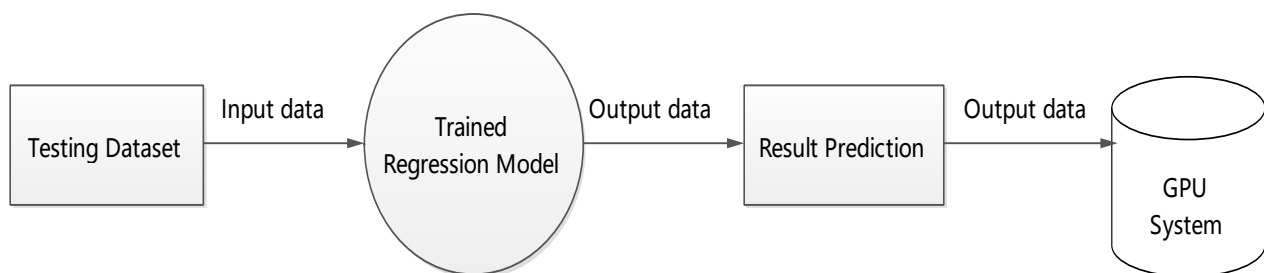
5.3.5 LEVEL 1



5.3.6 LEVEL 2



5.3.7 LEVEL 3



5.4 UML DIAGRAM

The Unified Modeling Language (UML) is a graphical language for visualizing, specifying, constructing and documenting the artifacts of a software-intensive system. The UML offers a standard way to write a system's blueprints, including conceptual things such as business processes and system functions as well as concrete things such as programming language, database schemas, and reusable software components. Here three UML diagrams are specified.

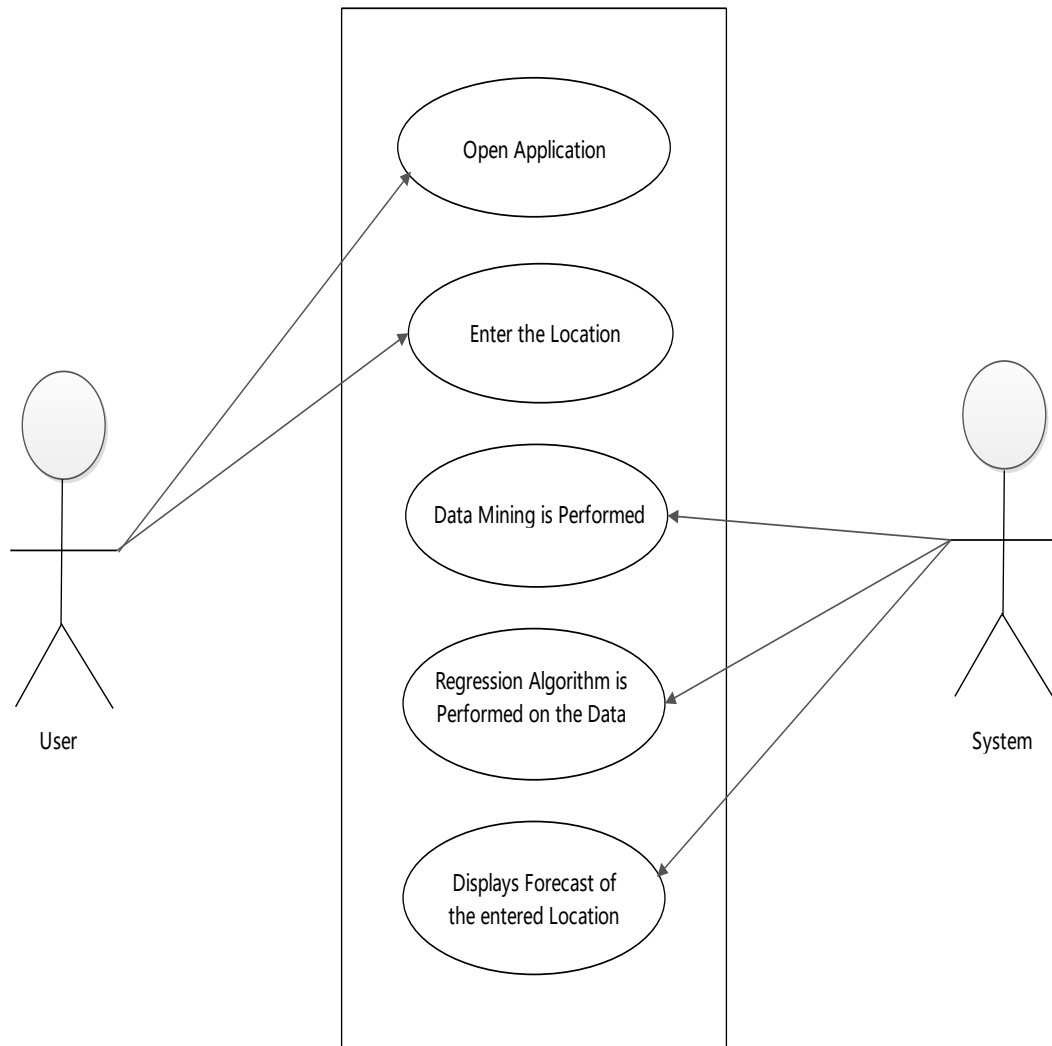
They are:

- ❖ Use Case diagram
- ❖ Sequence diagram
- ❖ Modular Diagram

5.4.1 USE CASE DIAGRAM

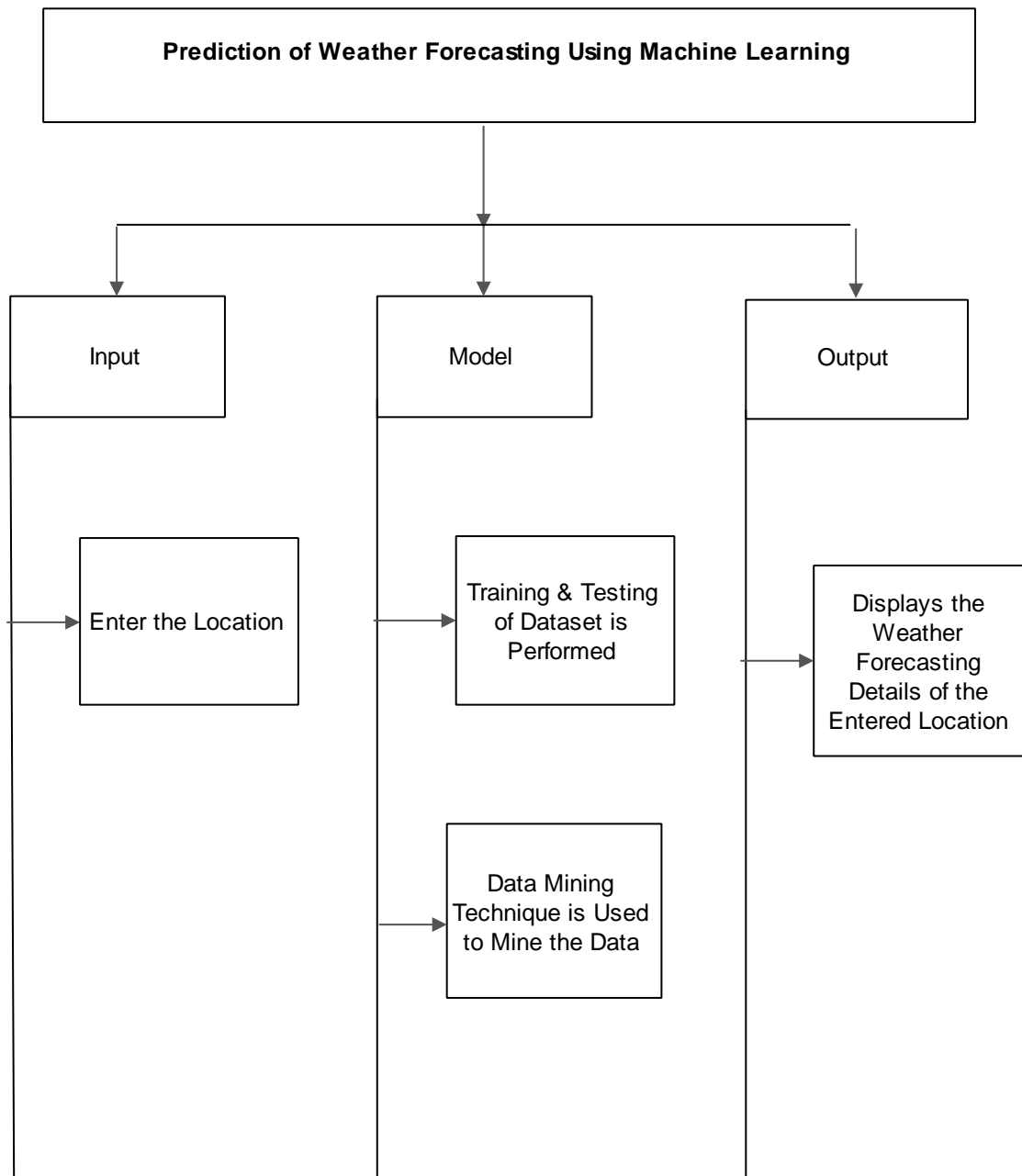
A use case is the set of scenarios that describing an interaction between a user and a system. A use case diagram displays relationship among actors and use cases. The two main components of a use diagram are use case and act. A user is an external view of the system that represents some action the user might perform in order to complete a task. The actor can be a human external system.

Prediction of weather forecasting using machine learning

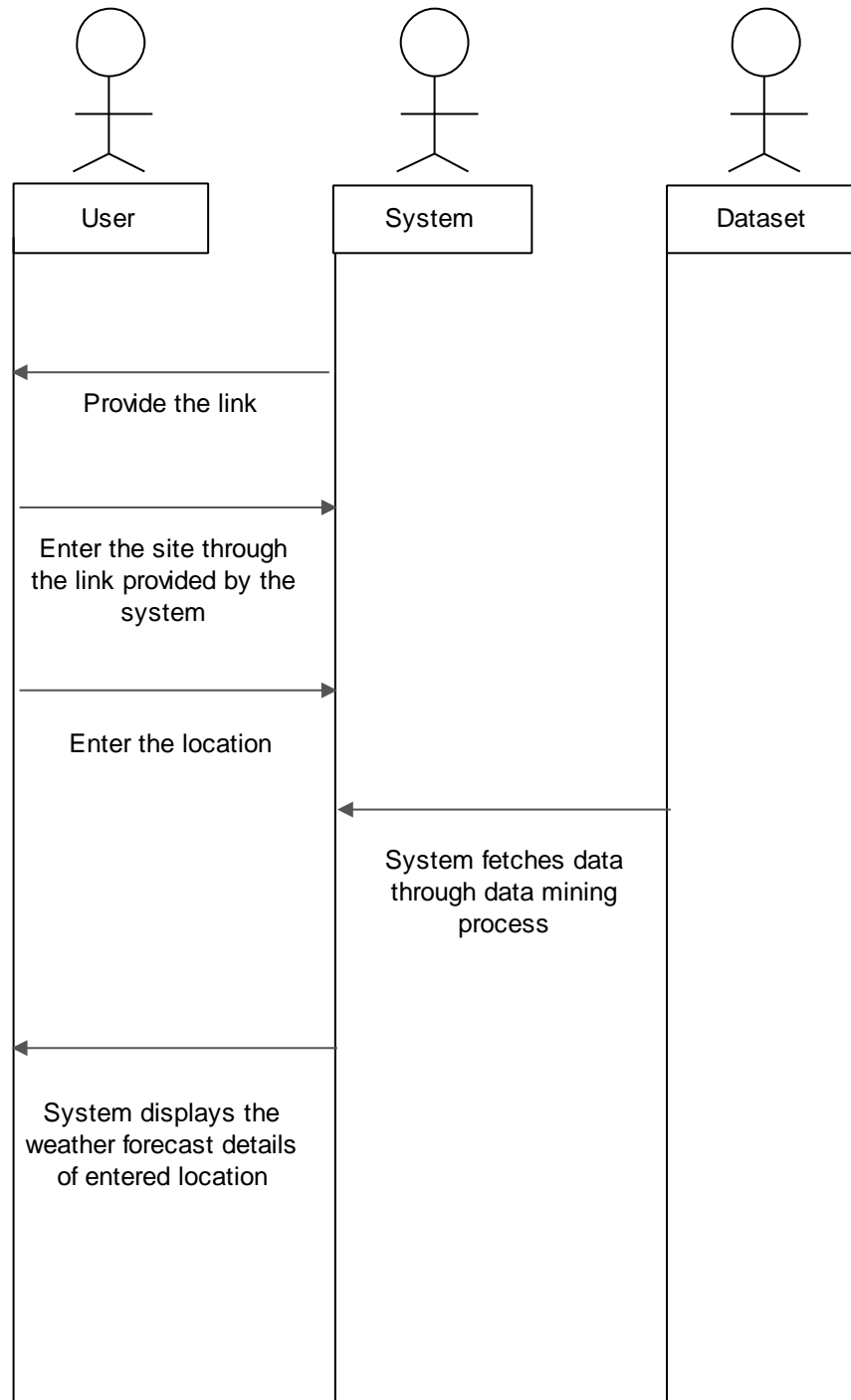


5.4.2 MODULAR DIAGRAM

Modular diagram is used to represent the modules in the software. In the diagram rectangles and arrows are used to represent the working of modules.



5.4.3 SEQUENCE DIAGRAM



CHAPTER 6

SYSTEM IMPLEMENTATION

6.1 INTRODUCTION

Implementation is the process of converting a new or revised system design into operation. It is the key stage in achieving a successful new system because, usually it reveals a lot of up and down in the user department. It must therefore be carefully planned and controlled. Apart from planning the two major tasks of preparing for implementation are education and training of users and testing of the system. Education of users should really take place much earlier in the project. i.e. when they are involved in the investigation and design work. Training has to be given to the staff regarding the new system. Once staff has been trained, the system can be tested. Implementation is the stage of the project where the theoretical design is turned into working system or it is the key stage in achieving a successful new system, because usually it involves a lot of up and down in the user department. Therefore it must be carefully planned and controlled. It can also be considered to be the most crucial stage in achieving a successful new system and in giving the user confidence that the new system will work and be effective.

Implementation is the final and important phase. It is the phase where theoretical design is turned into working system, which works for the user in the most effective manner. It involves careful planning, investigation of the present system and the constraints involved, user training, system testing and successful running of developed proposed system. The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system. The user tests the developed system and changes are made according to their needs. The testing phase involves the testing of a system using various kinds of data. This method also offers the greatest security.

Algorithm

6.2.1 Linear Regression Algorithm

Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting. Different regression models differ based on – the kind of relationship between dependent and independent variables, they are considering and the number of independent variables being used.

The equation has the form:

$$Y = a + b * X$$

Where Y is the dependent variable

X is the independent variable

b is the slope of the line and a is the y-intercept.

Step 1: Load the data into R. Follow these four steps for each dataset:

Step 2: Make sure your data meet the assumptions.

Step 3: Perform the linear regression analysis.

Step 4: Check for homiostatistics.

Step 5: Visualize the results with a graph.

Step 6: Report your results.

6.2.2 Decision Tree Regression Algorithm

Decision tree is a calculation that utilizes a tree like chart or model of Decisions and their potential results to foresee an official Decision, this calculation utilizes contingent control articulation. A Decision tree is a calculation for moving toward discrete-esteemed objective capacities, in which Decision tree is signified by an educated capacity. For inductive learning these sorts of calculations are popular and have been effectively applied to abroad scope of

errands. We offer name to another exchange that is whether it is genuine or misrepresentation for which class name is obscure and afterward exchange esteem is tried against the Decision tree, and after that from root hub to yield/class name for that exchange a way is followed. Decision guidelines decides the result of the substance of leaf hub. As the rules have the type of 'On the off chance that condition 1 and condition 2 yet not condition 3, at that point result. Decision tree assists with deciding the most exceedingly terrible, best and anticipated qualities for various situations, rearranged to comprehend and decipher and permits expansion of new potential situations.

step 1: Find the best attribute and place it on the root node of the tree.

step 2: Now, split the training set of the dataset into subsets. While

step3: Making the subset make sure that each subset of training dataset should have the same value for an attribute.

step4: Find leaf nodes in all branches by repeating 1 and 2 on each subset.

CHAPTER 7

SYSTEM TESTING

INTRODUCTION

Testing involves a series of operation of a system of application under controlled conditions and subsequently evaluating the result. The controlled condition should include both normal and abnormal conditions. It is planned and monitor for each testing level. The various testing performed are unit testing, integration testing, validation testing, output testing and system testing.

7.2 LEVELS OF TESTING

7.2.1 UNIT TESTING

A level of the software testing process where individual units of a software are tested. The purpose is to validate that each unit of the software performs as designed. The first level of testing, unit testing, is the most micro-level of testing. It involves testing individual modules or pieces of code to make sure each part or “unit” is correct. A “unit” can be a specific piece of functionality, a program, or a particular procedure within the application. Unit testing helps verify internal design and internal logic, internal paths, as well as error handling. The unit testing level includes a single type of testing; unit testing. Unit tests are done by the developer who wrote the code.

7.2.2 INTEGRATION TESTING

A level of the software testing process where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Integration testing is done after unit testing. This level tests how the units work together. Individual modules are combined and tested as a group. It's one thing if units work well on their own, but how do they perform together? Integration testing helps you determine that, and ensures your application runs efficiently. It identifies interface issues between modules.

There are a few techniques that can be used for conducting integration testing:

- Big Bang Testing
- Top Down Approach
- Bottom Up Approach

Big bang testing involves testing the entire set of integrated components together simultaneously. Because everything is integrated together and being tested at one time, this approach makes it difficult to identify the root cause of problems.

The top down approach starts by testing the top-most modules and gradually moving down to the lowest set of modules one-by-one.

The bottom up approach starts with testing the lowest units of the application and gradually moving up one-by-one.

7.2.3 SYSTEM TESTING

This level of testing is the first level that tests the entire application as a whole. It is often done in a mirrored production environment. This level of testing is actually a series of tests whose purpose is to test the application end-to-end. In this testing process where a complete, integrated system is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements.

System testing is particularly important because it verifies the technical, functional, and business requirements of the software. System testing is the last level of testing before the user tests the application.

There are dozens of types of system testing, including usability testing, regression testing and functional testing. This level of testing is typically done by the testing team and includes a combination of automated testing and manual testing.

7.2.4 ACCEPTANCE TESTING

A level of the software testing process where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery.

The final level of testing, acceptance testing, or UAT (user acceptance testing), determines whether or not the software is ready to be released. Let's face it, requirements change

through out the development process. It's important that the user verifies the business needs are met before the software is released into production.

Are the functional requirements met? Are the performance requirements met? These are the questions that are answered during acceptance testing level. UAT is the final say as to whether the application is ready for use in real life or not. This phase also involves change control managing requested modifications and new feature requests.

Acceptance testing should be done by the business user / end-user.

7.3 TESTING TECHNIQUES

7.3.1 BLACK-BOX TESTING

The technique of testing without having any knowledge of the interior workings of the application is called black-box testing. The tester is oblivious to the system architecture and does not have access to the source code. Typically, while performing a black-box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

7.3.2 WHITE-BOX TESTING

White-box testing is the detailed investigation of internal logic and structure of the code. White-box testing is also called glass testing or open-box testing. In order to perform whitebox testing on an application, a tester needs to know the internal workings of the code. The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

7.3.3 GREY-BOX TESTING

Grey-box testing is a technique to test the application with having a limited knowledge of the internal workings of an application. In software testing, the phrase the more you know, the better carries a lot of weight while testing an application. Mastering the domain of a system always gives the tester an edge over someone with limited domain knowledge. Unlike black-box testing, where the tester only tests the application's user interface; in grey-box testing, the tester.

has access to design documents and the database. Having this knowledge, a tester can prepare better test data and test scenarios while making a test plan.

7.4 TEST CASE AND OUTPUT

The test case is a document that describes an input, action, or event and an expected response to determine if a feature of an application is working correctly. A test case should contain particulars such as test case identifiers, test case, name, objectives, test conditions, input data requirements steps and expected results. This suggests the need for retesting and to discover the source of differences. The major document produced by the system analysis at the end of the system study stage. It provides complete details of the analyst's proposed solution to the problem outlined in terms of references. This is the description of the proposed new computer system in great deal; it specified how the system would do it. The system specification describes the hardware and software specification to develop the software.

7.4.1 Test cases

Project title : **Prediction of weather**

Software tool : Python

Test objectives : Predict the weather in selected city

Test data : City

Step No	Steps	Data	Expected Results	Actual Results
1	Enter a city	City name	Display Weather available on the city	Process successful
2	Enter a city	Invalid City name	Display the error message "weather not found"	Search again

CHAPTER 8

CONCLUSION AND FUTURE ENHANCEMENT

8.1 CONCLUSION

The weather prediction has become one of the most essential entity now a days. To improve the risk management systems and to know the weather in coming days in an automatic and in scientific way, many models have been emerging to assist in weather Prediction. linear regression were outperformed by professional weather forecasting services, although the discrepancy in their performance de- creased significantly for later days, indicating that over longer periods of time, our models may outperform pro- fessional ones. Linear regression proved to be a low bias, high variance model whereas functional regression proved to be a high bias, low variance model. Linear regression is inherently a high variance model as it is unstable to out- liers, so one way to improve the linear regression model is by collection of more data. This bias could be due to the design choice to forecast weather based upon the weather of the past two days, which may be too short to capture trends in weather that functional regression requires. If the forecast were instead based upon the weather of the past four or five days, the bias of the functional regression model could likely be reduced. However, this would require much more computation time along with retraining of the weight vector w , so this will be deferred to future work.

8.2 FUTURE ENHANCEMENT

Consequently, the model can be exploited without additional computational effort to perform accurate inferences in new scenarios with a wide range of data characteristics.

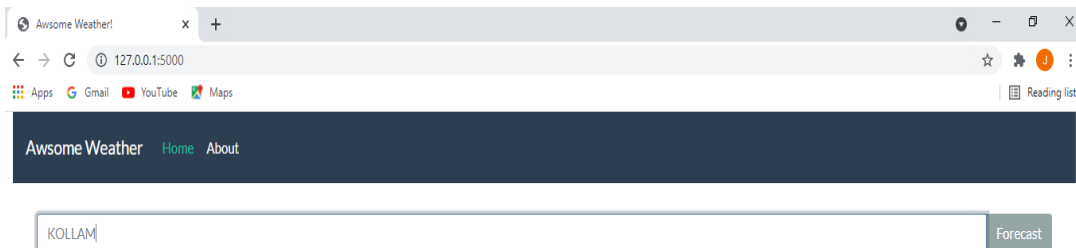
As future work, will try to decrease undesired side effect of the proposed model to gain less information loss.

APPENDICES

Appendix A

SCREENSHOTS

Home Page



Weather In Kollam

[Awsome Weather](#) [Home](#) [About](#)

Forecast for Kollam

Today	Tomorrow	Day after tomorrow
moderate rain temperature: 27.93°C	moderate rain temperature: 27.13°C	moderate rain temperature: 26.1°C

Search Again

Activate Windows
Go to Settings to activate Windows.

127.0.0.1:5000

Weather In Karavaloor

[Awsome Weather](#) [Home](#) [About](#)

Forecast for Karavaloor

Today	Tomorrow	Day after tomorrow
light rain temperature: 27.77°C	light rain temperature: 26.51°C	light rain temperature: 24.8°C

Search Again

Activate Windows
Go to Settings to activate Windows.

Weather In Kottarakara

Awsome Weather [Home](#) [About](#)

Forecast for Kottarakara

Today	Tomorrow	Day after tomorrow
moderate rain temperature: 27.86°C	moderate rain temperature: 26.46°C	light rain temperature: 24.69°C

[Search Again](#)

Activate Windows
Go to Settings to activate Windows.

Weather In Chadayamangalam

Awsome Weather [Home](#) [About](#)

Forecast for Chadayamangalam

Today	Tomorrow	Day after tomorrow
light rain temperature: 27.71°C	moderate rain temperature: 26.51°C	light rain temperature: 24.95°C

[Search Again](#)

Activate Windows
Go to Settings to activate Windows.

Weather In Pandalam

Awsome Weather [Home](#) [About](#)

Forecast for Pandalam

Today	Tomorrow	Day after tomorrow
moderate rain temperature: 26.25°C	heavy intensity rain temperature: 25.68°C	heavy intensity rain temperature: 24.59°C

[Search Again](#)

Activate Windows
Go to Settings to activate Windows.

Weather in Ayoor

Awsome Weather [Home](#) [About](#)

Forecast for Ayoor

Today	Tomorrow	Day after tomorrow
light rain temperature: 27.62°C	moderate rain temperature: 26.39°C	light rain temperature: 24.79°C

[Search Again](#)

Activate Windows
Go to Settings to activate Windows.

REFERENCES

- [1] N. Singh, S. Chaturvedi, and S. Akhter, “Weather Forecasting Using Machine Learning Algorithm,” *2019 Int. Conf. Signal Process. Commun. ICSC 2019*, pp. 171–174, 2019.
- [2] A. G. Salman, B. Kanigoro, and Y. Heryadi, “Weather forecasting using deep learning techniques,” *ICACSYS 2015 - 2015 Int. Conf. Adv. Comput. Sci. Inf. Syst. Proc.*, pp. 281–285, 2016.
- [3] W. Zhang, H. Zhang, J. Liu, K. Li, D. Yang, and H. Tian, “Weather prediction with multiclass support vector machines in the fault detection of photovoltaic system,” *IEEE/CAA J. Autom. Sin.*, vol. 4, no. 3, pp. 520–525, 2017.
- [4] Y. Beeharry, T. P. Fowdur, and J. A. Sunglee, “A Cloud-Based Real-Time Weather Forecasting Application,” pp. 294–297, 2019.
- [5] C. Kunjumon, S. S. Nair, S. Deepa Rajan, L. Padma Suresh, and S. L. Preetha, “Survey on Weather Forecasting Using Data Mining,” *Proc. IEEE Conf. Emerg. Devices Smart Syst. ICEDSS 2018*, no. March, pp. 262–264, 2018.

Websites:

- [1]. <https://www.learnpython.org>
- [2]. www.codecademy.com/learn/learn-python
- [3]. www.python.org
- [4]. w3schools.Com

