WEATHER FORECASTING SYSTEM USING GAUSSIAN NAÏVE BAYES

MAY THEINGI KYAW

M.C.Sc. **JUNE 2022**

WEATHER FORECASTING SYSTEM USING GAUSSIAN NAÏVE BAYES

By

MAY THEINGI KYAW

B.C.Sc.

A Dissertation Submitted in Partial Fulfillment of the Requirement for the Degree of Master of Computer Science (M.C.Sc.)

University of Computer Studies, Yangon

JUNE 2022

ACKNOWLEDGEMENTS

First of all, I would like to express my inmost gratitude to **Prof. Dr. Mie Mie Khin**, Rector, the University of Computer Studies, Yangon, for allowing me to develop this research and giving me excellent guidance during the period of my dissertation.

Secondly, I would like to express my deepest gratitude to my supervisor **Dr. Ah Nge Htwe**, Professor, Faculty of Computer Science, the University of Computer Studies, Yangon, for her caring and encouragement, and providing me with excellent ideas and guidance during the time of writing this dissertation.

Thirdly, I would like to express very special thanks to **Dr. Si Si Mar Win**, Professor, and **Dr. Tin Zar Taw**, Professor, Faculty of Computer Science, the University of Computer Studies, Yangon, as a Dean of Master's Course, for giving me the valuable guidance and suggestions during the development of this thesis.

In addition, I would like to acknowledge and special thanks to **Daw Aye Aye Khine**, Associate Professor, Head of the Department of English, the University of Computer Studies, Yangon. I would like to thank her for valuable supports and revising my dissertation from the language point of view.

Finally, I would like to extend my thanks to all my teachers who taught me throughout the master's degree course and my friends for their cooperation. The completion of my dissertation would not have been possible without the support and nurturing of my family.

STATEMENT OF ORIGINALITY

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

…… ……

Date May Theingi Kyaw

ABSTRACT

Forecasting is the process of estimation in unknown situations from the historical data. Weather forecasting is one of the applications to predict the atmosphere state for a future time and a given location as regards temperature, cloudiness, dryness, wind, rain and etc. Weather forecasting system is implemented as the responsive web application depends on OpenWeather API. The point of the proposed system is to classify the climate class using Gaussian Naïve Bayes based on weather information acquired from this API. The architecture of the proposed system consists of API from Open Weather, MySQL database for data storage of historical weather dataset, and PHP programming component. In this system, 63648 weather records are contained in the weather database and then 52608 records are used as training data and 11040 are testing.

The system has achieved nearly 89% accuracy using training data. For testing data, the system has achieved nearly 79% accuracy. According to the analysis outcomes, the proposed weather system can classify for climate status with the highest accuracy.

CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1
INTRODUCTION

Since ages, weather prediction remained the most thrilling and testing task for the specialists and meteorologists. Estimating climate unequivocally and precisely assists humankind with being ready for any regular catastrophes before time. Today, foreseeing weather conditions needs aptitude in different fields and furthermore includes processing complex numerical computations. The fast development in innovation has empowered researchers to precisely anticipate climate more. Precise forecast of climate boundaries is a troublesome errand because of the powerful idea of environment.

One of the intelligent approaches that have been proven in terms of predicting classification problems is Machine learning (ML). Using machine learning, weather prediction is less expensive, less time-consuming, convenient, real-time, and accurate. The conditions of climate estimation using techniques include doing apply many past weather data.

The proposed system intends to provide the user with the weather conditions of local city (Yangon) in Myanmar simultaneously in every time and any place.

1.1 Weather Forecasting

Weather estimation is the application of science and technology to prognosticate the atmospheric status for a given area and time frame. People have tried to foretell the climate informally for millennia and officially from way back. Weather prognosis are made by gathering information data about the present event of the atmosphere, land, and ocean and using weather station to propose how the atmospheric state will alter at a given place [22].

A change of works areas are dependent directly or indirectly on the status of weather. Agricultural and industrial sectors are chiefly dependent on weather state. Statuses of weather are also used to warn about natural catastrophe. This is why exact weather estimation is needed to command the tasks dependent on the weather or make determination for these tasks according to the weather outlook.

There are various techniques to forecast the weather conditions such as persistence, use of a barometer, looking at the sky, news casting, analog technique

and use of machine learning based forecasting models. Among them, the simplest way of outlook the weather is persistence, relies upon the present time's state to estimate the state the next day. Measurements of barometric pressure and the pressure tendency (the alteration of pressure time over) have been used in estimation since the late 19th century. Along with pressure tendency, the sky status is one of the most essential parameters used to foretell weather in mountainous regions. Thickening of cloudiness or the invasion of a higher cloud deck is indicative of rain in the near future. The estimation of the climate within the next six hours is concerned to as nowcasting. In this time range, it is possible to prognosticate little features such as individual showers and thunderstorms with fair truth, as well as other features too little to be decided by a model of computer. The analog method is a complex way of doing a prognosis, needing the prognosticator to recall an old weather event that is anticipated to be mimicked by a forthcoming event. What does it a hard method to utilize is that there is seldom an exact analog for an event in the future. Some call this type of foretell pattern acknowledgement [22].

The Machine Learning technique is the most robust technique for predicting weather forecasting. In past days, machine learning algorithms gave instructions to the system and gave results but now machine learning algorithms can directly give features and generate results for inputs automatically. Most of the work related to machine learning for agriculture either solves the purpose of cultivating a crop or suggests weather data based on statistical information. Climate status will be fluctuating every few hours and sometimes undergo forceful alternation. Cognizing the conditions of climate advance aids people in many ways by the smallest losses. The traditional climate estimation ways which make employ of satellite images and weather stations are highly prices because they consist of highly prices and complex ways. Many climate estimation techniques are not reliable and are less accurate local. Data is the essential in machine learning. Weather information is taken for the preparing process and then the training model is used to foretell the climate status to command distinct tasks. The state of weather prognosis using machine learning techniques include doing utilize of many past weather parameters. Therefore, a cheaper and better weather prediction system is necessary by using OpenWeather API based on machine learning algorithms [14].

1.2 Objectives of the Thesis

The main objective of thesis is to predict and classify a weather condition using Gaussian Naïve Bayes classifier based on localized weather dataset obtained from OpenWeather API. The additional objectives of the thesis are as follows:

- To collect and maintain the historical weather data of Yangon using OpenWeather API.
- To apply Gaussian Naïve Bayes algorithm for forecasting weather data.
- To support for weather stations and other in reality.
- To implement the weather forecasting system using web technology and MySQL database.
- To evaluate the accuracy of forecasted data using Gaussian Naïve Bayes classification algorithm with confusion matrix.

1.3 Problem Statement and Motivation of the Thesis

For agricultural development and disaster prevention, it is necessary to build a weather base station to monitor and predict the right weather conditions depending on temperature, humidity, pressure, clouds, visibility, dewpoint, and wind speed. In the low-income, underdeveloped areas of Myanmar, a low-cost weather forecasting system is needed to monitor and predict the right weather conditions required for agriculture. Farmers in Myanmar can improve their agriculture by knowing real-time weather parameters and predicting possible weather conditions. Therefore, it is necessary to build a system that is predictable as a meteorological station.

1.4 Contributions of the Thesis

The proposed system is low-cost weather prediction system for weather forecasting using responsive web design technology. The proposed weather forecasting system is very useful for Department of Meteorology and Hydrology to give accurate weather condition. The thesis contributions are as follows:

(i) Historical Dataset for local city (Yangon) from Myanmar is obtained from OpenWeather API.

(ii) Weather prediction system for weather forecasting based on Gaussian Naïve Bayes classifier combined with the responsive web technology is proposed.

1.5 Organization of the Thesis

This thesis is organized as five chapters, abstract, acknowledgment and references.

Weather prediction system based on machine learning is introduced in Chapter 1. This chapter also depicts the problem statement, motivation, contribution, aim, and objectives of the research work.

The fundamentals of the machine learning techniques, existing weather APIs and responsive web technology are introduced in Chapter 2. Among the respective types of machine learning classifier, Gaussian Naïve Bayes algorithm is briefly explained in this chapter. Related works to this research is also reviewed in this chapter.

Design and implementation of web-based weather forecasting system are presented in Chapter 3. First of all, the use case diagram of proposed system, the proposed system design overview, the system architecture, and the historical weather dataset structure are depicted in this chapter. And then, the implementation of programming modules for the proposed system is explicated with Graphical User Interfaces.

In Chapter 4, the experimental outcomes of proposed system are displayed with charts and tables. The calculations of precision, recall, F1-Score and overall accuracy for Gaussian Naïve Bayes classifier depend on weather forecasting system are presented in this chapter.

The conclusion of research work is described in Chapter 5. In this chapter, furthermore extensions that suggest some betterment which could be made are submitted. The system limitations are also described in this chapter.

CHAPTER 2
BACKGROUND THEORY AND RELATED WORKS

The weather is a natural calamity that always modifies with alters of unknown air parameters. Still, the average or mean status can be anticipated which ultimately gives the climate of a geographic region for ages consideration. The essential items that affect state of atmospheric are pressure, temperature, and dampness. Future weather determination depends on the historically gathered data is termed “Weather Estimation”. Climates foretell is a convenient instance for studying machine learning. Underdeveloped APIs for accessing obtainable parameters from meteorological institutes and other weather stations gives a copiousness data [15].

This chapter presents the various types of weather stations and using machine learning algorithms in weather forecasting. The different types of machine learning algorithms based on labeled or unlabeled dataset is discussed in this chapter. Among the various machine learning classifiers, three types of Naïve Bayes based classifiers, such as Multinomial, Bernoulli and Gaussian, are explained briefly as the background theory. And then, History API are presented to collect the historical weather dataset and to get access to climate parameters, such as temperature, dampness, wind speed, pressure, clouds, visibility and dew point. Responsive Web Design technology is also discussed in this chapter to develop web-based weather forecasting system. And finally, the past related works within the topic of weather prediction using machine learning modeling is reviewed.

2.1 Weather and Weather Station

Climate is the condition of the ambiance, depicting for instance the level to which it is hot or cold, wet, or dry, calm, or stormy, clear, or cloudy. Weather usually consists of temperature, humidity, clouds, visibility, dew point, pressure, and wind speed. A meteorological observation post is an aggregation of devices that measure state of atmospheric to assist study the climate of a precise location.

Most meteorological departments will measure temperature, dampness, barometric pressure, wind speed, visibility, dew point, and clouds. An official weather station is used for aviation, military, and meteorological resolves, while personal

home climate stations are for individual or, in some instances, research apply. Weather hobbyists, farmers, and schools are the most common owners of personal climate stations [7].

Figure 2.1 Weather Station with Instruments

2.1.1 Personal Weather Station

Personal weather stations are nowhere near as complicated and do not have the truth level that official climate stations. Nevertheless, more expensive models are able enough to be utilized in applications of research. Most will describe on temperature, dampness, barometric pressure, wind speed and direction, and rainfall. Some stations offer extra detectors that are utilitarian for specific applications. Lightning and UV/light detectors might be providing assistance for those who often take part in outside activities and land wet and leaf wet detectors are utilitarian for gardeners and farmers.

Personal stations are often linked to the Internet sharing parameters and monitor states remotely through platforms such as Climate Underground and Ambient Climate Organization. Their mensuration can also trigger mobile devices. Climate forecasters will even apply the information from personal camp. Individual stations can fill in the hole where official weather stations are not. The Cooperative Weather Observer Program (CWOP) is an organization of climate camp from volunteers that share climate lookout camp with the NWS [7].

2.1.2 Professional Weather Station

The professional climate camp means that it gives high rank and professional parameter that absolutely meets the International Meteorological Standards giving away, highly precise and real-time climate mensuration. The message received that these professional weather base camp provide with optional detectors are:

(a) Barometric Pressure
(b) dampness
(c) Temperature
(d) Precipitation
(e) Wind Speed
(f) Wind Direction

2.1.3 Usage of Weather Station

A weather camp can aid us prepare for the climate ahead and offer localized and sometimes the greater exact information than climate app that may be accepting data from a tower that is miles away from this position. Some of the likely reasons have been listed and why a meteorological observation post may be utilitarian below [6].

(a) Using it to acquire more around the atmospheric where are lived.
(b) Living in a distant area without a recognized meteorological observation post nearby.
(c) Being gardeners or farmers, it is necessary to recognize when to grow and harvest
(d) Doing the initiation smart home devices with the parameter.

If the business is atmospheric support, it can be optimized to run more expeditiously according to watch this status.

2.2 Machine Learning in Weather Forecasting

Weather condition determining is remained the most exciting and challenging task for the experts and climate prognosticator. Forecasting weather precisely and accurately helps mankind to be prepared for any natural calamities before time.

Today, predicting weather needs expertise in multiple fields and also regards as computing complex mathematical calculations. The occurring in a brief period of time growth in technology has enabled scientists to predict climate more exactly. Accurate prediction of weather data is a hard job because of the dynamic nature of atmosphere. Weather forecasting is the most common way of gathering information on air conditions, which enters the temperature, dampness, pressure, wind speed, etc. utilizing fast PCs, wired and remote detectors, satellites and climate radars [24]. With the advancement in data processing complicate arithmetical and statistical models has enabled us to obtain the best weather forecasting. A number of climate forecasting models are suggested by researchers to enhance verity of the models [18] [12] [5]. Many kinds of weather research and forecasting models (WRF), numerical weather prediction models (NWP), and automatic weather stations (AWS) are matured and applied to foretell climate more exactly [1]. Gigantic measure of climate informational index accessible in metrological focuses contains enormous volume of climate information which is utilized for the climate expectation.

From the beyond couples of many years, information mining strategies are generally broadly utilized for climate expectation and have shown a wonderful degree of precision and pertinence in expectation. Data mining is the investigation of how to decide concealed designs in the information to assist with pursuing ideal choices on PCs when the data set included is voluminous, difficult to portray precisely, and continually evolving. It sends strategies in light of AI, close by the ordinary techniques. These strategies can deliver choice or forecast models, in view of the huge volumes of real verifiable information. Along these lines, they address genuine proof based choice help [1]. Different order AI calculations are executed to anticipate the atmospheric conditions. Decision Trees, Artificial Neural Networks (ANN), Naive Bayes Networks, Support Vector Machines (SVM), Fuzzy Logic, Genetic Algorithms are some of the ordinarily used data mining techniques that are mostly applied for weather estimation.

Based on the data type, i.e., labeled, or unlabeled data, there are various kinds of ML algorithms, usually, these algorithms are collected into two categories: unsupervised and supervised learning [13].

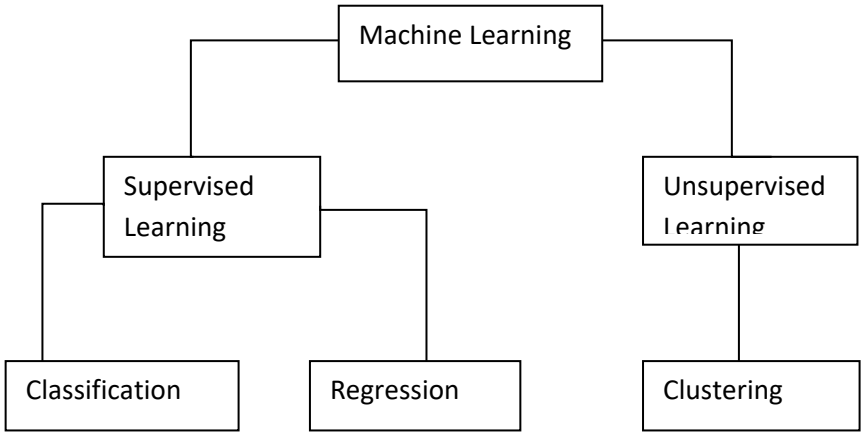


Figure 2.2 Machine Learning Algorithms

2.2.1 Supervised Learning

Supervised learning is the machine learning job of learning a function that maps an input to an output depends on example input-output pairs. It infers a function from labeled preparing data including a set of preparing instances. In supervised learning, each instance is a pair including an input object and the desired output value.

This set of algorithms uses preparing data to generate a function that maps the inputs to desired outputs (also called labels). For example, in the issues of classification, the system looks at instance data and uses it to arrive at a function mapping input data into classes. Artificial neural organization, radial basis function organization, and decision trees are forms of supervised learning [13].

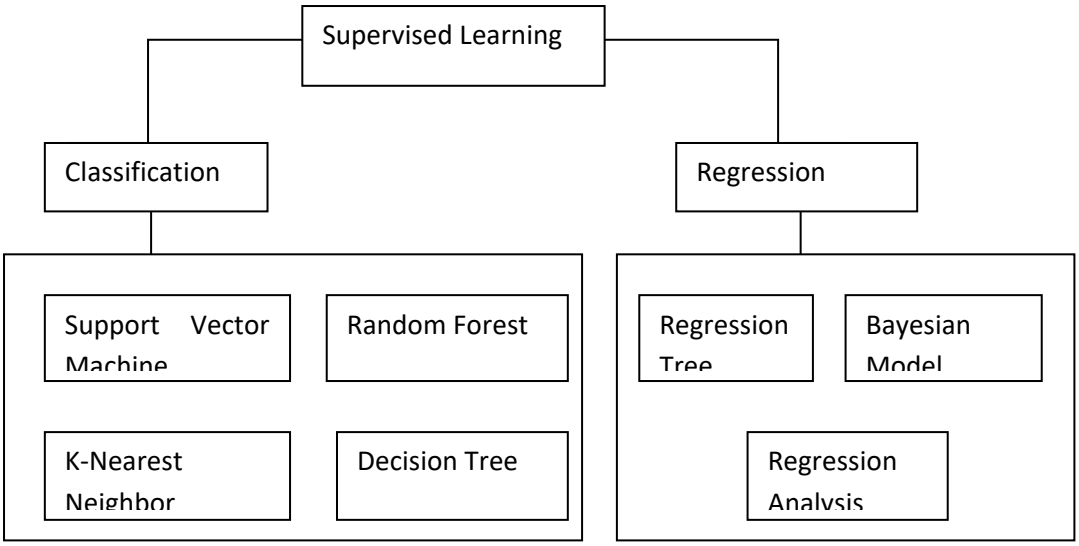


Figure 2.3 Supervised Learning Algorithms

2.2.2 Unsupervised Learning

Unsupervised learning is a type of machine learning in which the calculation is not provided with any pre-assigned labels or scores for the preparing data. As a result, unsupervised learning algorithms must first self-discover any naturally occurring patterns in those preparing parameters. This set of calculation works without previously labeled data.

The chief purpose of these calculations is to chance the common forms in antecedently unseen data. Clustering is the most popular form of unsupervised learning. Hidden Markov models and self-organizing maps are other forms of Unsupervised Learning. Unsupervised learning algorithms include clustering, anomaly detection, neural networks, etc. These calculations find hidden patterns or parameter groupings without the acquired for human intervention [13].

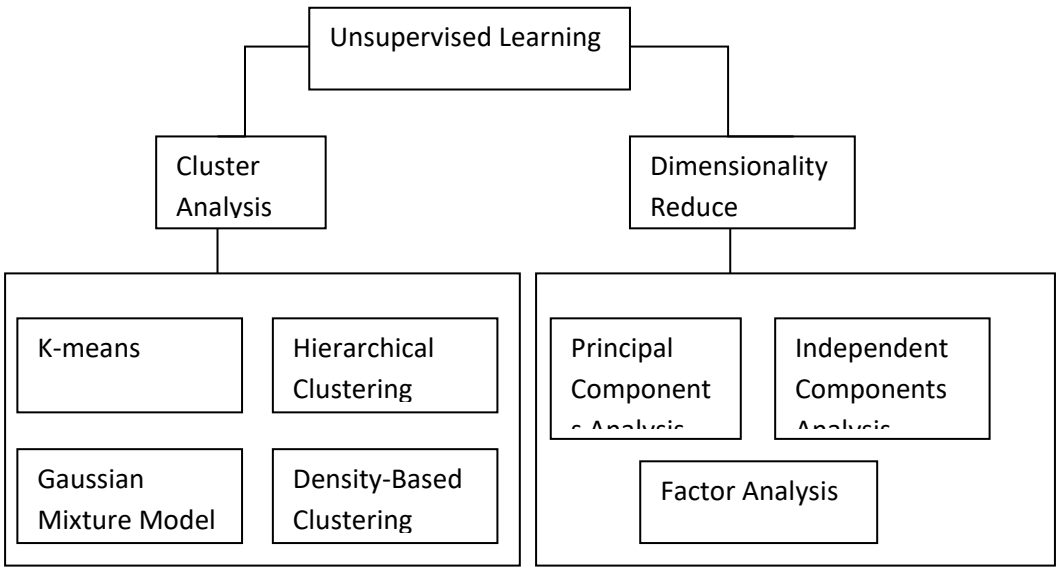


Figure 2.4 Unsupervised Learning Algorithms

2.2.3 Semi-supervised Learning

Semi-supervised machine learning is a compound of supervised and unsupervised machine learning methods. In semi-supervised learning, a calculation studies from a dataset that contains both labeled and unlabeled information, normally mostly unlabeled. Artificial neural organizations are exceedingly famous in the field of foretelling in wireless organizations. The other techniques like decision trees and

unsupervised learning are utilized much lesser. Exploratory proves that applying a compound of techniques instead of a single one provides the best outcomes [14].

2.3 Naïve Bayes Classifier

Naïve Bayes classification is a form of supervised learning. The Naïve Bayes classifier expects that the worth of one element is autonomous of the worth of some other component. Innocent Bayes classifiers need preparing information to assess the boundaries expected for characterization. Because of straightforward plan and application, Naïve Bayes classifiers can be appropriate in some genuine situations. The technique is leisurely to understand when depicted utilizing binary or categorical input parameters. Several of Naïve Bayes Algorithm is:

(a) Multinomial Naïve Bayes
(b) Bernoulli Naïve Bayes
(c) Gaussian Naïve Bayes

2.3.1 Multinomial Naïve Bayes

Multinomial Naive Bayes calculation is a probabilistic learning technique that is mostly applied in Natural Language Processing (NLP). The calculation is depending on the Bayes theorem and foretells the tag of a text such as a piece of email or newspaper article. With a multinomial event model, samples (feature vectors) represent the frequencies which precise events have been generated by a multinomial $(p_1,\ldots,p_n)$ where p_i is the probability that event i occurs (or K such multinomial in the multiclass case). A feature vector $x = (x_1,\ldots,x_n)$ is then a histogram, with x_i counting the number of times event i was observed in a particular case. This is the event model typically utilized for document classification, with events representing the occurrence of a word in a single document. The likelihood of observing a histogram x is given by

$$p(x \mid C_k) = \frac{(\sum_{i=1}^{n} x_i)!}{\prod_{i=1}^{n} x_i!} \prod_{i=1}^{n} p_{ki}^{x_i}$$

where, 2.1

$p(x \mid C_k)$ = likelihood of observing a histogram x,

x_i = number of times event i was observed in a particular instance,

p_{ki} = probability that event i occurs

2.3.2 Bernoulli Naïve Bayes

In the multivariate Bernoulli event model, attributes are autonomous Booleans (binary variables) depicting inputs. Like the multinomial model, this model is famous for document classification jobs, where binary term occurrence elements are utilized instead of term frequencies [3]. If x_i is a Boolean showing the occurrence or absence of the ith term from the vocabulary, then the likelihood of a document given a class C_k is given by

$$p(x \mid C_k) = \prod_{i=1}^{n} p_{ki}^{x_i} \ (1 - p_{ki})^{(1-x_i)}$$

where, 2.2

p_{ki} = the probability of class C_k ,

x_i = Boolean expression the occurrence of absence of the ith term from the vocabulary.

This event model is especially famous for classifying short document. It has the benefit of explicitly modeling the terms absence. Naïve Bayes classifier with a Bernoulli event model is different as a multinomial Naïve Bayes classifier with frequency reckon shortened to one [3].

2.3.3. Gaussian Naïve Bayes

The Gaussian probability density function can be utilized to make expectations by subbing the boundaries with the new info worth of the variable and accordingly, the Gaussian capacity will give a gauge for the new information worth's likelihood.

The Gaussian Naïve Bayes classifier is a fast and straightforward classifier method that functions admirably without an excess of exertion and a decent degree of precision. Gaussian Naïve Bayes is not difficult to fabricate and especially helpful for enormous datasets. It is particularly utilized for mathematical information (persistent and discretize information). For instance, the training parameters consist of a persistent attribute, x. The information is first divided by the label, and then the mean and variance of x is calculated in each label [3] as the following 2.3 and 2.4 equations:

$$\mu_k = \frac{(\sum x_i)}{n}$$

where, 2.3

μ_k = mean of the values in x associated with class C_k,

n = number of data points,

$\sum x_i$ = the sum of data points.

$$\sigma_k^2 = \sum (x_i - \mu)^2 / (n - 1)$$

where, 2.4

σ_k^2 = variance of the values in x associated with class C_k,

x_i = data points,

n = number of data points,

μ = mean of the values in x.

Let μ_k be the mean of the values in x associated with class C_k, and let σ^2 be the variance of the values in x associated with class C_k. Then the probability density of x for a given class C_k can be computed [3] by following 2.5 equation.

$$p(x \mid C_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(x-\mu_k)^2}{2\sigma_k^2}}$$

where, 2.5

$p(x \mid C_k)$ = likelihood which is the probability of predictor (attribute) given class.

The advantages of Gaussian Naïve Bayes are:

(i) It is a fast and easy model gives highly suitable results.
(ii) There is no need to spend much time for training.
(iii) It provides better and accurate performance.
(iv) It works well with large data.
(v) It has no zero frequency due to continuous data.

2.4 Weather APIs

API is an acronym for Application Programming Interface. An API call is the process of a client application submitting a demand to an API and that API retrieving the demanding parameter from the outside host or program and sending it back to the client. It can be understood as a composition of rules that enables us to access an external service on the web through our systems. Thus, an API determines and sets certain formats wherein we can access the service and the data to and from a model. Through a programming language perspective such as Python, an API is considered as a data source available on the web which can be accessed through particular libraries of a programming language [8].

Good weather APIs provide both historical weather data and forecast data via an easy-to-use, well-defined programming interface. The best APIs have dozens of weather measures, near-real-time current conditions reporting, and decades of worldwide historical weather reports. Ideally, both historical and forecast look-ups would be combined into the same API entry point with the addition of an ultra-long-range forecast based on climate statistics. This single-entry point makes it easy for anyone writing a script, coding, an app, or loading a database to get instant access to the exact weather data that they need from a global database containing hundreds of millions of records. Of course, the pricing for this API should be cheap enough that anyone can get access and initial users should be able to start their weather project entirely for free [4].

OpenWeather [9] platform is a set of elegant and widely recognizable APIs. Powered by convolutional machine learning solutions, it is capable of sending all the climate information essential for determination for any area on the globe.

2.4.1 Historical Weather Data API

OpenWeather provide hourly historical weather data for any location on the globe via History API [9].

2.5 Responsive Web Technology for Weather Applications

Nowadays users obtain the same website from desktop computers, laptops, mobile phones, iPhones, iPads, Blackberries, notebooks, feed readers and even smart TVs. Each platform shows the same page in not the same feel from the others based on its size and viewing capabilities.

Whenever a user enters an internet site, the client looks for a user-friendly interface, quick access to his/her needs and a comfortable content view without the need to worry about how they are accessing it [19]. On the other hand, web designers and programmers have to guarantee, as much as possible, to provide such an experience to all their clients and from all the different internet-accessing technologies.

Nowadays, with the big betterment in mobile phones and its ever-growing increase in their popularity, as published reports indicate that at least 65% of people aged 18-29 are accessing the internet using mobile devices [2], therefore more responsive or adaptive website design techniques are needed and used to ensure the best possible viewing experience to the users.

Responsive Web Design (RWD) is a name given to the set of techniques used to develop one single website which adapts itself on unknown devices and is capable of reshaping itself based on various screen sizes, resolutions and orientations from the largest devices like the internet TV to the smallest ones on mobile devices, another name used to depict it is “Adaptive Web Design” which as its name indicates relate to the techniques utilized in a website to enable it to adapt to different viewing devices [20][23]. The unknown devices with various screen sizes and resolutions are presented in Figure 2.5 and 2.6.

Figure 2.5 Different Devices with Various Screen Sizes

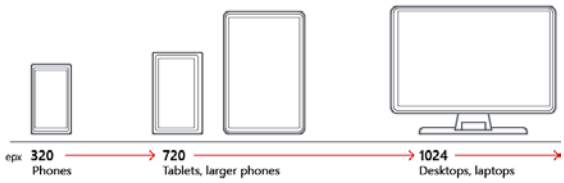


Figure 2.6 Different Devices with Various Screen Resolutions

Responsive web design (RWD) is a web development approaching that creates dynamic changes to the appearance of a website, based on the screen size and orientation of the device being utilized to view it. The same HTML is served to all devices, applying CSS (which determines the layout of webpage) to change the appearance of the page. Bootstrap CSS is the most famous framework for building mobile responsive web applications. PHP is one of the most famous scripting languages to develop the web-based applications applying APIs.

Therefore, using PHP and Responsive Web Design approach provides to design and implement the proposed system as a web-based weather application for the multiple of devices available to users.

2.6 Related Works

R. Siddhant and D. Shaba [19] depicted "Weather Forecasting Using Data Mining". Expectation model is utilized for precipitation forecast. Among of the 15 elements, 5 highlights are applied for estimating. Climate information gathered from NOAA (National Oceanic Atmospheric Administration). KNN and Gaussian Naïve Bayes are utilized to gauge the precipitation. Mathematical traits are utilized and result for downpour or not. Gaussian Naïve Bayes are more precise than KNN.

Nikam, B. Valmik and B. B. Meshram [23] introduced "Modeling Rainfall Prediction Using Data Mining Method - A Bayesian Approach". Information is acquired from Indian Meteorological Department. Among of the 36 qualities, just 7 ascribes which are generally applicable to precipitation forecast are thought of, and to decide downpour or not. Guileless Bayesian is utilized for precipitation expectation. Mathematical quality is utilized. This framework works with effective precision

Santhanam et al [21] advised “Neural Network based model for weather forecasting”. Analysts and researchers are involving Neural Network for climate

expectation because of his effortlessness, strength and adequacy. In this work the creators have carried out and analyzed the exhibitions of engendering brain organization (BPN) and revolutionary premise worked brain organization (RBF). For this exploration work, the total climate information of a decade is gathered from meteoric division, Kanyakumari, Tamil Nadu, India. The outcomes proved that extreme premise worked brain organization (RBF) have better exactness, quicker and more dependable for climate expectation. The prescient precision of RBF was 88.49% which makes it more helpful for quick continuous weather conditions gauging.

KavithaRani et al [10] suggested “a novel rainfall prediction model using hybrid classifier” executing fake honey bee state calculation is coordinated effort with the hereditary calculation for preparing the feed forward brain organization. For the work the creators gathered the genuine climate informational index from Rayalaseema, Aandhra and Telangana areas of India. The precipitation expectation is finished utilizing the crossover classifier. From the exploratory outcomes, it was found that the proposed crossover classifier has preferable execution and prescient capacity over Artificial Bee Colony with Neural Organization.

2.7 Summary

This chapter submits the brief of theory background about weather forecasting, machine learning algorithms and weather APIs. The previous works based on machine learning approach which relate to this weather forecasting research are reviewed in detail in this chapter. Weather forecasting is the most common way of gathering information on air conditions, which records the temperature, dampness, precipitation, wind speed and its heading utilizing fast PCs, wired and remote sensors, satellites and climate radars. In this research work, weather APIs are proposed to use to collect the historical weather data. In this chapter, various types of machine learning algorithms are reviewed and three groups of Naïve Bayes machine learning classifier are described in detail. Gaussian Naïve Bayes is proposed to classify the weather class based on the trained historical weather dataset. Responsive Web Design approach is described to pattern and implement this system as a web-based application for the multiple of devices available to users.

CHAPTER 3
THE SYSTEM DESIGN AND IMPLEMENTATION OF WEATHER FORECASTING SYSTEM

Based on the aim and objectives, the researcher will design and implement a weather prediction system that is truly useful locally depend on machine learning classifier. The detailed implementation of Gaussian Naïve Bayes based weather forecasting system is submitted in this chapter. Design and use case diagrams of the system, database structure of proposed system also included in this chapter. Collecting of historical weather dataset by using OpenWeather APIs are also discussed in detail. The system of logical architecture and implementation of programming components are also explicated in this chapter. Finally, it shows a system of graphical user interface with step by step explication.

3.1 Overview Design of the System

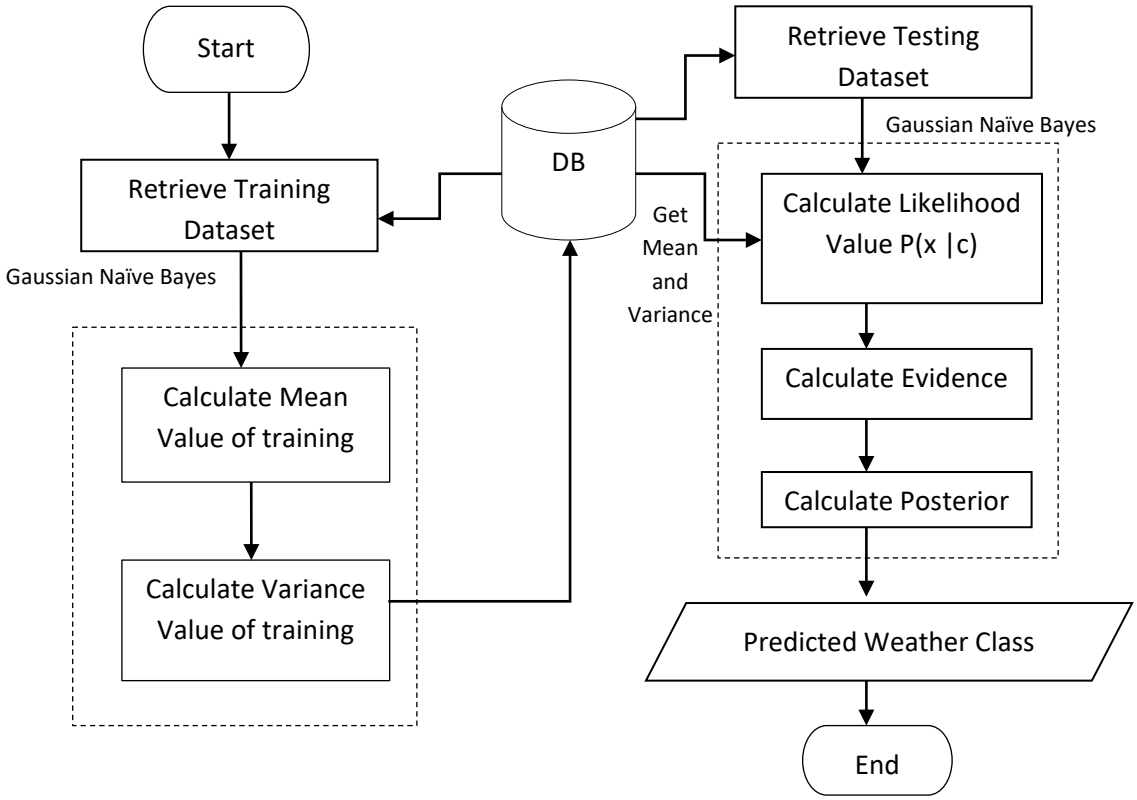


Figure 3.1 Overview Design of the System

The overview design diagram of system is shown in Figure 3.1. The proposed system is implemented as the web-based weather forecasting system by Gaussian Naïve Bayes classifier. The chief point of the proposed system is the Gaussian Naïve Bayes classification for weather class based on weather parameters. In this system, there are two phases: training and testing, and four main steps: get weather conditions and historical weather information from APIs, train weather dataset, classification of weather class by Gaussian Naïve Bayes and evaluation of classification results.

In the first step of training phases, collecting of historical weather data, which consists of the seven main attributes, like temperature, humidity, wind speed, pressure, cloud, visibility and dew-point, is performed. In this system History API from OpenWeather is used to get historical weather data for Yangon location. In the second step, the collected weather data are stored in MySQL weather database and trained with Gaussian Naïve Bayes classifier for weather prediction process.

The testing phase consists of three steps: getting the values of weather parameters from database, classification of weather class based on parameters' values and evaluation of classification results. In the first step, the numerical values for parameters: temperature, humidity, wind speed, pressure, cloud condition, visibility and dew_point, are obtained from database. In the second step, based on these parameters' values, weather forecasting is performed by classification with Gaussian Naïve Bayes. In the final step, the predicted weather class is compared with actual weather class received from database and evaluation results for prediction is saved into MySQL database.

3.2 Use Case Diagram of the System

The primary aim of this use-case diagram is to envision the functional requirements of proposed system, consisting of the relationship of "actors" (human beings who will connect with the system) to indispensable processes, as well as the relationships among different use cases. The main functional processes and their relationship to actors of the proposed weather forecasting system are shown in Figure 3.2 by use-case diagram. There is one role of actor; User can view historical weather data and Gaussian Naïve Bayes model. The user can also predict weather condition and view experimental result for training and testing.

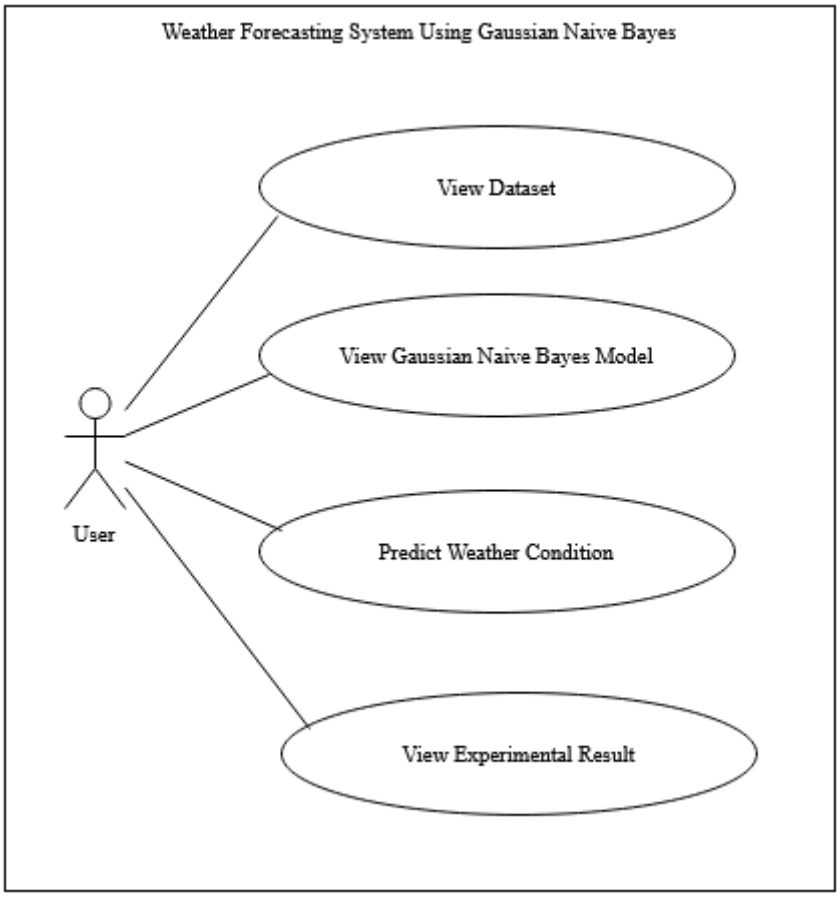


Figure 3.2 Use-case Diagram of the System

3.3 Implementation of the system

The web-based weather prediction system for weather forecasting is implemented which depends on Model View Controller (MVC) by utilizing the PHP based responsive web technology and OpenWeather APIs. The system of logical architecture is shown in Figure 3.3. The architecture of the proposed system includes API from OpenWeather, MySQL database for data storage of historical weather dataset, and two programming components. All programming components are implemented by using PHP programming language and jQuery. All functions for the proposed system can be grouped into two modules: API Request and Application Module, and Classification Module.

History API from OpenWeather provides historical weather data for Yangon and these API provides weather conditions with forecast.

The main functions of the API request and application module are the collecting of historical weather data and the display of weather. Collected historical

weather data for six years consists of 52608 datasets is saved to database as a training dataset by this module.

Gaussian Naïve Bayes classification module plays in main role of the weather prediction system for weather forecasting. The training of historical data, classification of weather conditions and evaluation of classification results are the main functions of this module. API request module provides the historical weather data and values of weather parameters to Gaussian Naïve Bayes classification module and classification results weather conditions are received from this module as a response. In this system architecture, MySQL database is used to store the historical weather dataset, classification results.

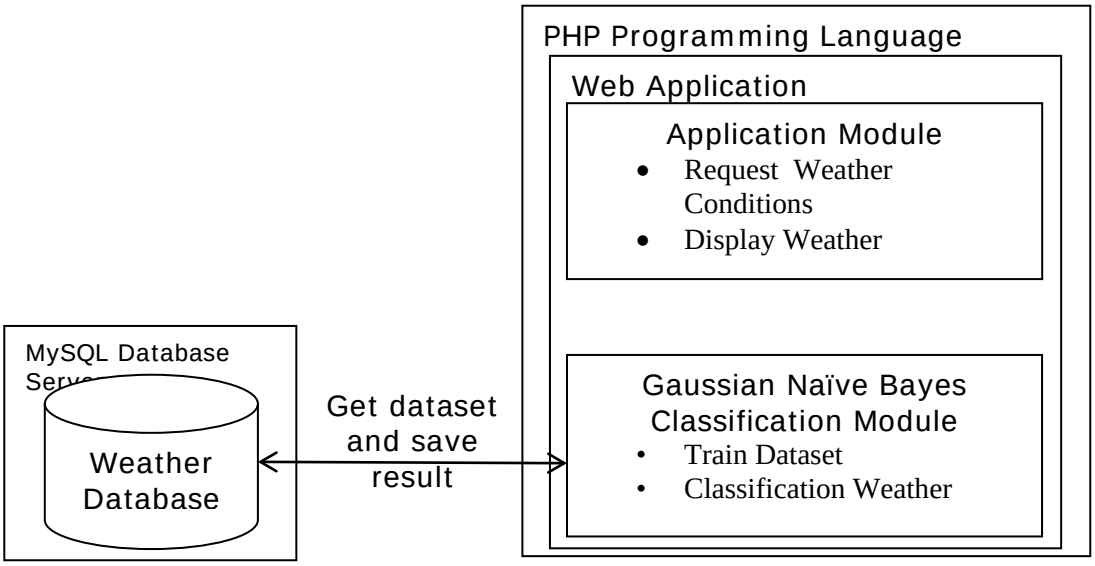


Figure 3.3 Architecture of the System

3.3.1 Implementation of Weather Database in MySQL

In this section the schema for weather database is presented. There are three tables. Schema is designed and implemented by MySQL RDMBS for saving all of the information of historical weather data and the results of Gaussian Naïve Bayes classification.

Training Dataset	
PK	id
	date_time temp humidity windy pressure cloud visibility dew_point actual _class

Testing Dataset	
PK	id
	date_time temp humidity windy pressure cloud visibility dew_point actual _class

Training Model	
PK	id
	tempClearmean tempClearvar humClearmean humClearvar windClearmean windClearmean . . . dewSmokemean dewSmokevar

Figure 3.4 Schema of the Weather Database

3.3.2 Example Calculation using Gaussian Naïve Bayes

In this section Gaussian Naïve Bayes classification for weather prediction is explained with sample historical weather data. The sample dataset with values of six weather parameters for 26 is described in Table 3.1.

Table 3.1 Sample Historical Weather Data

Temp	Humidity	Wind Speed	Cloud	Pressure	Visibility	Dew_ Point	Class
20.2	93	1	1	1013	6000	19.03	Clear
20.47	94	1	1	1012	6000	19.47	Clear
21.89	73	1.5	20	1010	7000	16.83	Clouds
23.48	64	1.5	20	1010	7000	16.28	Clouds
25.01	88	2.6	20	1007	6000	22.88	Clouds
25.5	83	2.6	1	1009	7000	22.4	Clear
26.91	94	1.5	40	1009	6000	25.86	Clouds
30.78	65	3.1	40	1013	6000	23.44	Clouds
27.15	94	1.5	40	1011	7000	26.1	Rain
27.58	94	1.5	40	1011	7000	26.53	Rain
27.75	88	5.1	90	1006	6000	25.58	Rain
30.43	72	3.6	40	1007	7000	24.81	Rain
26.04	94	2.1	20	1003	7000	25	Clouds
26.61	94	2.1	40	1003	7000	25.56	Clouds
26.43	85	0.5	40	1009	6000	23.7	Haze
25.62	93	0.5	40	1010	6000	24.4	Haze
27.14	93	2.6	40	1007	7000	25.91	Drizzle
27.47	90	2.6	40	1009	7000	25.68	Drizzle
31.86	94	1.5	90	1003	5000	30.77	Thunderstorm
29.54	87	1.5	40	1006	8000	27.14	Thunderstorm
25.14	100	1.5	75	1010	500	25.14	Fog
25.46	98	1.5	75	1010	500	25.12	Fog
26.78	89	2.1	20	1009	5000	24.81	Mist
26.13	94	2.1	20	1010	5000	25.09	Mist
36.04	56	1.5	1009	40	5000	25.87	Smoke
38.13	40	1	1019	40	7000	22.16	Smoke

There are seven weather attributes in this sample dataset. The description of the weather attributes are as follows:

(a) Temp : temperature. Unit of this parameter is Celsius.

(b) Humidity : humidity in percentage (%).

(c) Wind_Speed : wind speed. The unit of this parameter is meter per seconds (m/s).

(d) Pressure : atmospheric pressure. (hpa)

(e) Cloud : cloudiness in percentage (%).

(f) Visibility : average visibility. (metres)

(g) Dew_Point : atmospheric pressure below which water droplets begin to condense and dew can form. (kelvin)

These are nine class labels as follows:

(a) Clear

(b) Clouds

(c) Rain

(d) Thunderstorm

(e) Drizzle

(f) Mist

(g) Fog

(h) Haze

(i) Smoke

3.3.2.1 Mean and Variance Calculation

According to the equation 2.3 and 2.4 the mean and variance values of all attributes for each class is calculated as follows:

"Temp" Attribute:

- for class "Clear"

  Mean

  = (20.2+20.47+25.5) / 3

  = 22.0567

Variance

$= ((20.2-22.0567)^2 + (20.47-22.0567)^2 + (25.5-22.0567)^2) / (3-1)$

$= 538.3874$

- for class “Clouds”

  Mean

  $= (21.89+23.48+25.01+26.91+30.78+26.04+26.61) / 7$

  $= 25.8171$

  Variance

  $= ((21.89-25.8171)^2+(23.48-25.8171)^2+(25.01-25.8171)^2+(26.91-25.8171)^2+(30.78-25.8171)^2+(26.04-25.8171)^2+(26.61-25.8171)^2)/ (7-1)$

  $= 75.1964$

- for class “Rain”

  Mean

  $= (27.15+27.58+27.75+30.43) / 4$

  $= 28.2275$

  Variance

  $= ((27.15-28.2275)^2+(27.58-28.2275)^2+(27.75-28.2275)^2+(30.43-28.2275)^2) / (4-1)$

  $= 145.6369$

- for class “Thunderstorm”

  Mean

  $= (31.86+29.54) / 2$

  $= 30.7$

  Variance

  $= ((31.86-30.7)^2+(29.54-30.7)^2) / (2-1)$

  $= 736.17$

- for class “Haze”

  Mean

  $= (26.43+25.62) / 2$

  $= 26.025$

  Variance

  $= ((26.43-26.025)^2+(25.62-26.025)^2) / (2-1)$

  $= 438.0453$

- for class “Drizzle”

  Mean

  = (27.14+27.47) / 2

  = 27.305

  Variance

  = $((27.14-27.305)^2+(27.47-27.305)^2) / (2-1)$

  = 406.6853

- for class “Fog”

  Mean

  = (25.14+25.46) / 2

  = 25.3

  Variance

  = $((25.14-25.3)^2+(25.46-25.3)^2) / (2-1)$

  = 493.602

- for class “Mist”

  Mean

  = (26.78+26.13) / 2

  = 26.455

  Variance

  = $((26.78-26.455)^2+(26.13-26.455)^2) / (2-1)$

  = 418.0043

- for class “Smoke”

  Mean

  = (36.04+38.13) / 2

  = 37.085

  Variance

  = $((36.04-37.085)^2+(38.13-37.085)^2) / (2-1)$

  = 2979.412

As a calculation result the mean and variance values of weather attributes for each weather class are obtained as shown in Table 3.2

Table 3.2 Mean and Variance Results of Training Sample Dataset

Class	Clear		Clouds		Rain	
Attributes	**Mean**	**Variance**	**Mean**	**Variance**	**Mean**	**Variance**
Temp	22.057	538.387	25.817	75.196	28.228	145.637
Humidity	90	2990.5	81.7143	932.5204	87	1809.6667
Wind_Speed	1.5333	14.141	2.057	4.203	2.925	17.182
Pressure	1011.333	943671.278	1007.86	312283.112	1008.75	625714.375
Cloud	1	1049039.5	28.5714	326367.534	52.5	622950.8333
Visibility	6333.33	41861111.1	6571.43	15045918.4	6750	32375000
Dew_Point	20.3	311.196	22.264	56.44	25.755	109.837

Class	Thunderstorm		Drizzle		Haze	
Attributes	**Mean**	**Variance**	**Mean**	**Variance**	**Mean**	**Variance**
Temp	30.7	736.17	27.305	406.685	26.025	438.045
Humidity	90.5	6118.5	91.5	6432.5	89	5745
Wind_Speed	1.5	28.96	2.6	37.1	0.5	76.16
Pressure	1004.5	186118.5	1008	1874247	1009.5	1880068.5
Cloud	65	1834015	40	1911815	40	1911815
Visibility	6500	88000000	7000	109500000	6000	79500000
Dew_Point	28.955	877.80	25.795	333.08	24.05	254.816

Class	Fog		Mist		Smoke	
Attributes	**Mean**	**Variance**	**Mean**	**Variance**	**Mean**	**Variance**
Temp	25.3	493.602	26.455	418.007	37.085	2979.412
Humidity	99	10445	91.5	6432.5	48	40841
Wind_Speed	1.5	28.96	2.1	25.241	1.25	35.885
Pressure	1010	1882035	1009.5	1880068.5	40	22517815
Cloud	75	1811995	20	1997455	1014	22913203
Visibility	500	844000000	5000	101500000	6000	79500000
Dew_Point	25.13	284.581	24.95	275.408	24.015	79500000

3.3.2.2 Evidence and Posterior Calculation

In testing of Gaussian Naïve Bayes classifier with sample data, first of all the probability of tested attributes for all possible class must be calculated by equation 3.3. The sample testing data is shown in Table 2.3. The step by step calculation of probability for all attributes is as follows:

Table 3.3 Sample Testing Data

Temperature	Humidity	WindSpeed	Cloud	Pressure	Visibility	Dew_Point
31.02	74	2.06	60	1015	7000	25.84

“Clear” class:

p(Clear)= 1 / 9 = 0.111

p(Temp|Clear)=1 / sqrt(2*3.14*538.387) * exp(-(31.02−22.057)^2/(2*538.387))
=0.01595

p(Humidity|Clear) = 1 / sqrt(2*3.14*2990.5) * exp(-(74−90)^2/(2*2990.5))
=0.0069

p(WindSpeed|Clear)=1 / sqrt(2*3.14*14.141) * exp(-(2.06−1.53)^2/(2*14.141))
=0.1051

p(Pressure|Clear)=1/sqrt(2*3.14*943671.278)*exp((1015−1011.333)^2/(2*943
671.278))
= 0.00041

p(Cloud|Clear) = 1 / sqrt(2*3.14*1049039.5) * exp(-(60−1)^2/(2*1049039.5))
= 0.00039

p(Visibility|Clear)=1/sqrt(2*3.14*41861111.1)*exp((7000−6333.33)^2/(2*4186
1111.1))
= 6.1334E-05

p(Dew_Point|Clear)=1/sqrt(2*3.14*311.196)*exp((25.84−20.3)^2/(2*311.196))
= 0.0215

“Clouds” class:

p(Clouds)= 1 / 9 = 0.111

p(Temp|Clouds)=1 / sqrt(2*3.14*75.196) * exp(-(31.02−25.817)^2/(2*75.196))
=0.03842

p(Humidity|Clouds)=1/sqrt(2*3.14*932.520)*exp((74−81.7143)^2/(2*932.520)
)
=0.0127

p(WindSpeed|Clouds)=1 / sqrt(2*3.14*4.203) * exp(-(2.06−2.057)^2/(2*4.203))
=0.1946

p(Pressure|Clouds)=1/sqrt(2*3.14*312283.112)*exp((1015−1007.86)^2/(2*312
283.112))
= 0.00017

p(Cloud|Clouds)=1/sqrt(2*3.14*326367.534)*exp((60−28.5714)^2/(2*326367.5
34))
= 0.00016

p(Visibility|Clouds)=1/sqrt(2*3.14*15045918.4)*exp((7000−6571.43)^2/(2*15
045918.4))
= 0.0001

p(Dew_Point|Clouds)=1/sqrt(2*3.14*56.44)*exp((25.84−22.264)^2/(2*56.44))
= 0.0474

“Rain” class:

p(Rain)= 1 / 9 = 0.111

p(Temp|Rain)= 1 / sqrt(2*3.14*145.637) * exp(-(31.02−28.228)^2/(2*145.637))
=0.0322

p(Humidity|Rain)=1/ sqrt(2*3.14*1809.6667) * exp(-(74−87)^2/(2*1809.6667))
=0.0089

p(WindSpeed|Rain)=1/ sqrt(2*3.14*17.182) * exp(-(2.06−2.925)^2/(2*17.182))
=0.0942

p(Pressure|Rain)=1/sqrt(2*3.14*625714.375)*exp((1015−1008.75)^2/(2*62571
4.375))
= 0.00024

p(Cloud|Rain)=1/sqrt(2*3.14*622950.8333)*exp((60−52.5)^2/(2*622950.8333)
)
= 0.00024

p(Visibility|Rain)=1/sqrt(2*3.14*3237500)*exp((7000−6750)^2/(2*3237500))
= 67.00464E-05

p(Dew_Point|Rain)=1/sqrt(2*3.14*109.84)*exp((25.84−25.75)^2/(2*109.84))
= 0.0381

“Thunderstorm” class:

p(Thunderstorm)= 1 / 9 = 0.111

p(Temp|Thunderstorm)=1/sqrt(2*3.14*736.17)*exp((31.02−30.7)^2/(2*736.17)
)
=0.0147

p(Humidity|Thunderstorm)=1/sqrt(2*3.14*6118.5)*exp((74−90.5)^2/(2*6118.5
))
=0.0049

p(WindSpeed|Thunderstorm)=1/sqrt(2*3.14*28.96)*exp((2.06−1.5)^2/(2*28.96
))
=0.07373

p(Pressure|Thunderstorm)=1/sqrt(2*3.14*186118.5)*exp((1015−1004.5)^2/(2*
186118.5))
= 0.00023

p(Cloud|Thunderstorm)=1/sqrt(2*3.14*1834015)*exp((60−65)^2/(2*1834015))
= 0.00023

p(Visibility|Thunderstorm)=1/sqrt(2*3.14*88000000)*exp((7000−6500)^2/(2*88000000))
= 4.2467E-05

p(Dew_Point|Thunderstorm)=1/sqrt(2*3.14*877.80)*exp((25.84−28.955)^2/(2*877.80))
= 0.0134

"Haze" class:

p(Haze)= 1 / 9 = 0.111

p(Temp|Haze)=1 / sqrt(2*3.14*438.045) * exp(-(31.02−26.025)^2/(2*438.045))
=0.0185

p(Humidity| Haze) = 1 / sqrt(2*3.14*5745) * exp(-(74−89)^2/(2*5745))
=0.0052

p(WindSpeed| Haze) = 1 / sqrt(2*3.14*76.16) * exp(-(2.06−0.5)^2/(2*76.16))
=0.0449

p(Pressure|Haze)=1/sqrt(2*3.14*1880068.5)*exp((1015−1009.5)^2/(2*1880068.5))
= 0.0002

p(Cloud| Haze) = 1 / sqrt(2*3.14*1911815) * exp(-(60−40)^2/(2*1911815))
= 0.00023

p(Visibility|Haze)=1/sqrt(2*3.14*79500000)*exp((7000−6000)^2/(2*79500000))
= 4.44626E-05

p(Dew_Point|Haze)=1/sqrt(2*3.14*254.816)*exp((25.84−24.05)^2/(2*254.816))
= 0.0248

"Drizzle" class:

p(Drizzle)= 1 / 9 = 0.111

p(Temp|Drizzle)=1/sqrt(2*3.14*406.685)* exp(-(31.02−27.305)^2/(2*406.685))
=0.0195

p(Humidity| Drizzle) = 1 / sqrt(2*3.14*6432.5) * exp(-(74−91.5)^2/(2*6432.5))
=0.0049

p(WindSpeed|Drizzle)=1/ sqrt(2*3.14*37.1) * exp(-(2.06−2.6)^2/(2*37.1))

=0.0652

p(Pressure|Drizzle)=1/sqrt(2*3.14*1874247)*exp((1015−1008)^2/(2*1874247))

= 0.00023

p(Cloud| Drizzle) = 1 / sqrt(2*3.14*1911815) * exp(-(60−40)^2/(2*1911815))

= 0.00023

p(Visibility|Drizzle)=1/sqrt(2*3.14*109500000)*exp((7000−7000)^2/(2*10950 0000))

= 3.81E-05

p(Dew_Point|Drizzle)=1/sqrt(2*3.14*333.08)*exp((25.84−25.795)^2/(2*333.08))

= 0.0219

"Fog" class:

p(Clear)= 1 / 9 = 0.111

p(Temp| Fog) = 1 / sqrt(2*3.14*493.602) * exp(-(31.02−25.3)^2/(2*493.602))

=0.0173

p(Humidity| Fog) = 1 / sqrt(2*3.14*10445) * exp(-(74−99)^2/(2*10445))

=0.0038

p(WindSpeed| Fog) = 1 / sqrt(2*3.14*28.96) * exp(-(2.06−1.5)^2/(2*28.96))

=0.0737

p(Pressure|Fog)=1/sqrt(2*3.14*1882035)*exp((1015−1010)^2/(2*1882035))

= 0.0002

p(Cloud| Fog) = 1 / sqrt(2*3.14*1811995) * exp(-(60−75)^2/(2*1811995))

= 0.00023

p(Visibility|Fog)=1/sqrt(2*3.14*844000000)*exp((7000−500)^2/(2*844000000))

= 1.33927E-05

p(Dew_Point|Fog)=1/sqrt(2*3.14*284.581)*exp((25.84−25.13)^2/(2*284.581))

= 0.0236

"Mist" class:

p(Mist)= 1 / 9 = 0.111

p(Temp|Mist)= 1 / sqrt(2*3.14*418.007) * exp(-(31.02−26.455)^2/(2*418.007))

=0.019

p(Humidity| Mist) = 1 / sqrt(2*3.14*6432.5) * exp(-(74−91.5)^2/(2*6432.5))

=0.0049

p(WindSpeed| Mist) = 1 / sqrt(2*3.14*25.241) * exp(-(2.06−2.1)^2/(2*25.241))
=0.0794

p(Pressure|Mist)=1/sqrt(2*3.14*1880068.5)*exp((1015−1009.5)^2/(2*1880068.5))
= 0.00023

p(Cloud| Mist) = 1 / sqrt(2*3.14*1997455) * exp(-(60−20)^2/(2*1997455))
= 0.00022

p(Visibility|Mist)=1/sqrt(2*3.14*101500000)*exp((7000−5000)^2/(2*101500000))
= 3.88E-05

p(Dew_Point|Mist)=1/sqrt(2*3.14*275.408)*exp((25.84−24.95)^2/(2*275.408))
= 0.024

"Smoke" class:

p(Smoke)= 1 / 9 = 0.111

p(Temp|Smoke)=1/sqrt(2*3.14*2979.412)*exp((31.02−37.085)^2/(2*2979.412))
=0.0073

p(Humidity| Smoke) = 1 / sqrt(2*3.14*40841) * exp(-(74−48)^2/(2*40841))
=0.0019

p(WindSpeed|Smoke)=1/sqrt(2*3.14*35.885) * exp(-(2.06−1.25)^2/(2*35.885))
=0.0659

p(Pressure|Smoke)=1/sqrt(2*3.14*22517815)*exp((1015−40)^2/(2*22517815))
= 8.407E-05

p(Cloud|Smoke)=1/sqrt(2*3.14*22913203)* exp(-(60−1014)^2/(2*22913203))
= 8.3343E-05

p(Visibility|Smoke)=1/sqrt(2*3.14*79500000)*exp((7000−6000)^2/(2*79500000))
= 4.4463E-05

p(Dew_Point|Smoke)=1/sqrt(2*3.14*79500000)*exp((25.84−24.015)^2/(2*79500000))
= 4.4743E-05

According to the probability of attributes for all class, the evidence is calculated by follows:

Evidence = p(Clear)*p(Temp|Clear)*p(Humidity|Clear)*p(WindSpeed|Clear)* p(Pressure|Clear)*p(Cloud|Clear)*p(Visibility|Clear)*p(Dew_Point|Clear) +p(Clouds)*p(Temp|Clouds)*p(Humidity|Clouds)*p(WindSpeed|Clouds)* p(Pressure|Clouds)*p(Cloud|Clouds)*p(Visibility|Clouds)*p(Dew_Point|Cloud) + p(Rain)*p(Temp|Rain)*p(Humidity|Rain)*p(WindSpeed|Rain)* p(Pressure|Rain)*p(Cloud|Rain)*p(Visibility|Rain)* p(Dew_Point|Rain)+ p(Thunderstorm)*p(Temp| Thunderstorm)*p(Humidity| Thunderstorm)*p(WindSpeed| Thunderstorm)* p(Pressure| Thunderstorm)*p(Cloud| Thunderstorm)*p(Visibility| Thunderstorm)*p(Dew_Point| Thunderstorm) + p(Drizzle)*p(Temp| Drizzle)*p(Humidity| Drizzle)*p(WindSpeed| Drizzle)* p(Pressure| Drizzle)*p(Cloud| Drizzle)*p(Visibility| Drizzle)*p(Dew_Point| Drizzle) + p(Haze)*p(Temp| Haze)*p(Humidity| Haze)*p(WindSpeed| Haze)* p(Pressure| Haze)*p(Cloud| Haze)*p(Visibility| Haze)* p(Dew_Point| Haze)+ p(Fog)*p(Temp| Fog)*p(Humidity| Fog)*p(WindSpeed| Fog)* p(Pressure| Fog)*p(Cloud| Fog)*p(Visibility| Fog)*p(Dew_Point| Fog)+p(Mist)*p(Temp| Mist)*p(Humidity| Mist)*p(WindSpeed| Mist)*p(Pressure| Mist)*p(Cloud| Mist)*p(Visibility| Mist)*p(Dew_Point| Mist) + p(Smoke)*p(Temp| Smoke)*p(Humidity| Smoke)*p(WindSpeed| Smoke)* p(Pressure| Smoke)*p(Cloud| Smoke)*p(Visibility| Smoke)* p(Dew_Point| Smoke)

= ((0.111*0.01595*0.0069*0.1051*0.00041*0.00039*6.1334E-05*0.0215)+(0.111*0.03842*0.0127*0.1946*0.00017*0.00016*0.0001*0.0474)+(0.111*0.0322*0.0089*0.0942*0.00024*0.00024*67.00464E-05*0.0381)+(0.111*0.0147*0.0049*0.07373*0.00023*0.00023*4.2467E-05*0.0134)+(0.111*0.0185*0.0052*0.0449*0.0002*0.00023*4.44626E-05*0.0248)+(0.111*0.0195*0.0049*0.0652*0.00023*0.00023*3.81E-05*0.0219)+(0.111*0.0173*0.0038*0.0737*0.0002*0.00023*1.33927E-05*0.0236)+(0.111*0.019*0.0049*0.0794*0.00023*0.00022*3.88E-05*0.024)+(0.111*0.0073*0.0019*0.0659*8.407E-05*8.3343E-05*4.4463E-05*4.4743E-05))

= 2.05841E-18

After the evidence is calculate, the posterior for each class is computed by follows:

Posterior(Clear) = (p(Clear)*p(Temp|Clear)*p(Humidity|Clear)*p(WindSpeed|Clear)* p(Pressure|Clear)*p(Cloud|Clear)*p(Visibility|Clear)*p(Dew_Point |Clear)) / evidence

= (0.111*0.01595*0.0069*0.1051*0.00041*0.00039*6.1334E-05*0.0215)/ 2.05841E-18

= 0.0506574

Posterior(Cloud)=(p(Clouds)*p(Temp|Clouds)*p(Humidity|Clouds)*p(WindSpeed|Clouds)*p(Pressure|Clouds)*p(Cloud|Clouds)*p(Visibility|Clouds)*p (Dew_Point|Cloud)) / evidence

=(0.111*0.03842*0.0127*0.1946*0.00017*0.00016*0.0001*0.0474) / 2.05841E-18

= 0.659526

Posterior(Rain) = (p(Rain)*p(Temp|Rain)*p(Humidity|Rain)*p(WindSpeed|Rain)* p(Pressure|Rain)*p(Cloud|Rain)*p(Visibility|Rain)* p(Dew_Point|Rain)) / evidence

= (0.111*0.0322*0.0089*0.0942*0.00024*0.00024*67.00464E-05*0.0381) / 2.05841E-18

=0.23048

Posterior(Thunderstorm) = (p(Thunderstorm)*p(Temp| Thunderstorm)*p(Humidity| Thunderstorm)*p(WindSpeed| Thunderstorm)* p(Pressure| Thunderstorm)*p(Cloud| Thunderstorm)*p(Visibility| Thunderstorm)*p(Dew_Point| Thunderstorm)) / evidence

= (0.111*0.0147*0.0049*0.07373*0.00023*0.00023*4.2467E-05*0.0134)/ 2.05841E-18

= 0.00879

Posterior(Drizzle)=(p(Drizzle)*p(Temp| Drizzle)*p(Humidity| Drizzle)*p(WindSpeed| Drizzle)* p(Pressure| Drizzle)*p(Cloud| Drizzle)*p(Visibility| Drizzle)*p(Dew_Point| Drizzle)) / evidence

=(0.111*0.0185*0.0052*0.0449*0.0002*0.00023*4.44626E-05*0.0248) / 2.05841E-18

= 0.01430

Posterior(Haze) = (p(Haze)*p(Temp| Haze)*p(Humidity| Haze)*p(WindSpeed| Haze)* p(Pressure| Haze)*p(Cloud| Haze)*p(Visibility| Haze)* p(Dew_Point| Haze)) / evidence

= (0.111*0.0195*0.0049*0.0652*0.00023*0.00023*3.81E-05*0.0219) / 2.05841E-18

=0.01321

Posterior(Fog) = (p(Fog)*p(Temp| Fog)*p(Humidity| Fog)*p(WindSpeed| Fog)* p(Pressure| Fog)*p(Cloud| Fog)*p(Visibility| Fog)*p(Dew_Point| Fog)) / evidence

= (0.111*0.0173*0.0038*0.0737*0.0002*0.00023*1.33927E-05*0.0236)/ 2.05841E-18

= 0.004404

Posterior(Mist)=(p(Mist)*p(Temp| Mist)*p(Humidity| Mist)*p(WindSpeed| Mist)*p(Pressure| Mist)*p(Cloud| Mist)*p(Visibility| Mist)*p(Dew_Point| Mist)) / evidence

= (0.111*0.019*0.0049*0.0794*0.00023*0.00022*3.88E-05*0.024) / 2.05841E-18

= 0.01862

Posterior(Smoke) = (p(Smoke)*p(Temp| Smoke)*p(Humidity| Smoke)*p(WindSpeed| Smoke)* p(Pressure| Smoke)*p(Cloud| Smoke)*p(Visibility| Smoke)* p(Dew_Point| Smoke)) / evidence

= (0.111*0.0073*0.0019*0.0659*8.407E-05*8.3343E-05*4.4463E-05*4.4743E-05) / 2.05841E-18

=7.06109E-07

Since the posterior numerator of “Clouds” is highest score, the predicted weather class for testing data, which shown in Table 2.3, is “Clouds”.

3.3.3 Implementation of APIs Request Module and Web Application

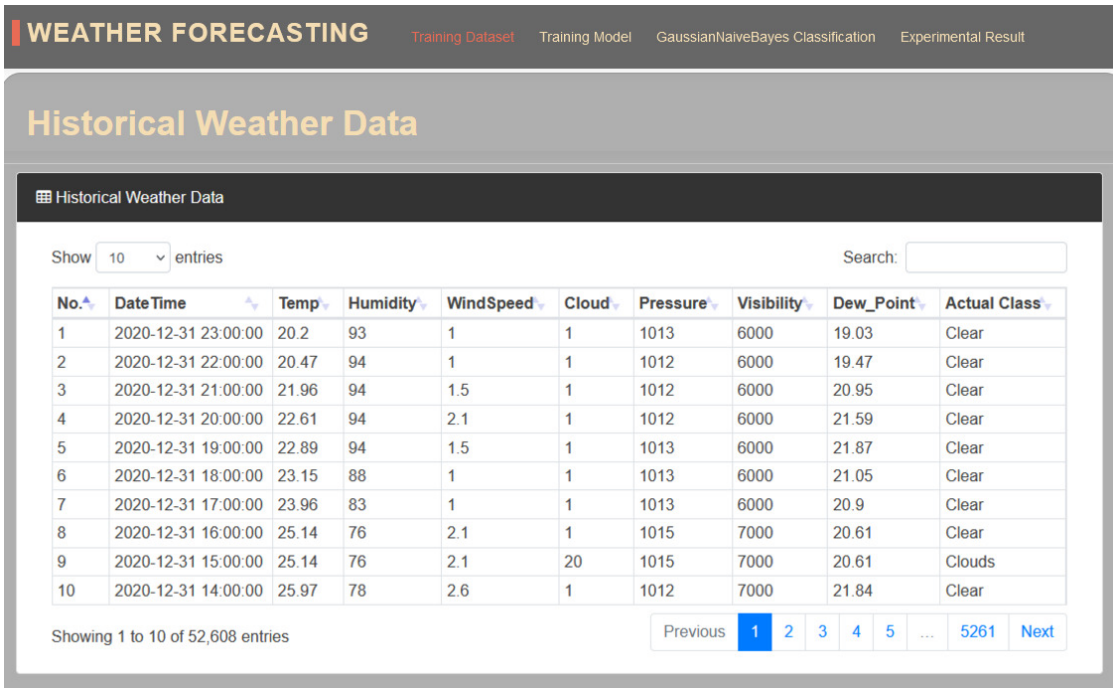
The API Request and Application Module of the proposed system are implemented by using PHP programming language and jQuery language. This request is done by using jQuery language in JS files and all web pages of the application are designed by PHP programming language based on Bootstrap 4.0 responsive web framework.

There are five JS files. Among them “train.js” file is used in API request module. This module consists of functions for collecting historical weather data, getting the weather conditions for weather prediction. These functions are as follows:

- train.js: JS file to get model for Gaussian Naïve Bayes based on historical weather data.

The user interface is designed and implemented like a web application in PHP. Type of role for the actor: User Role is defined in web applications. The users can view for weather conditions and test the weather classification process. This application consists of four menus: Train Dataset, Gaussian Naïve Bayes Classification, Experimental Result, and Dataset.

After clicking Dataset, the webpage of the “Dataset” menu can be described as shown in Figure 3.5.



No.	DateTime	Temp	Humidity	WindSpeed	Cloud	Pressure	Visibility	Dew_Point	Actual Class
1	2020-12-31 23:00:00	20.2	93	1	1	1013	6000	19.03	Clear
2	2020-12-31 22:00:00	20.47	94	1	1	1012	6000	19.47	Clear
3	2020-12-31 21:00:00	21.96	94	1.5	1	1012	6000	20.95	Clear
4	2020-12-31 20:00:00	22.61	94	2.1	1	1012	6000	21.59	Clear
5	2020-12-31 19:00:00	22.89	94	1.5	1	1013	6000	21.87	Clear
6	2020-12-31 18:00:00	23.15	88	1	1	1013	6000	21.05	Clear
7	2020-12-31 17:00:00	23.96	83	1	1	1013	6000	20.9	Clear
8	2020-12-31 16:00:00	25.14	76	2.1	1	1015	7000	20.61	Clear
9	2020-12-31 15:00:00	25.14	76	2.1	20	1015	7000	20.61	Clouds
10	2020-12-31 14:00:00	25.97	78	2.6	1	1012	7000	21.84	Clear

Figure 3.5 Dataset Page

In this weather forecasting web application, all the pages can be accessed by user. The home page of weather forecast web application is shown in Figure 3.6.

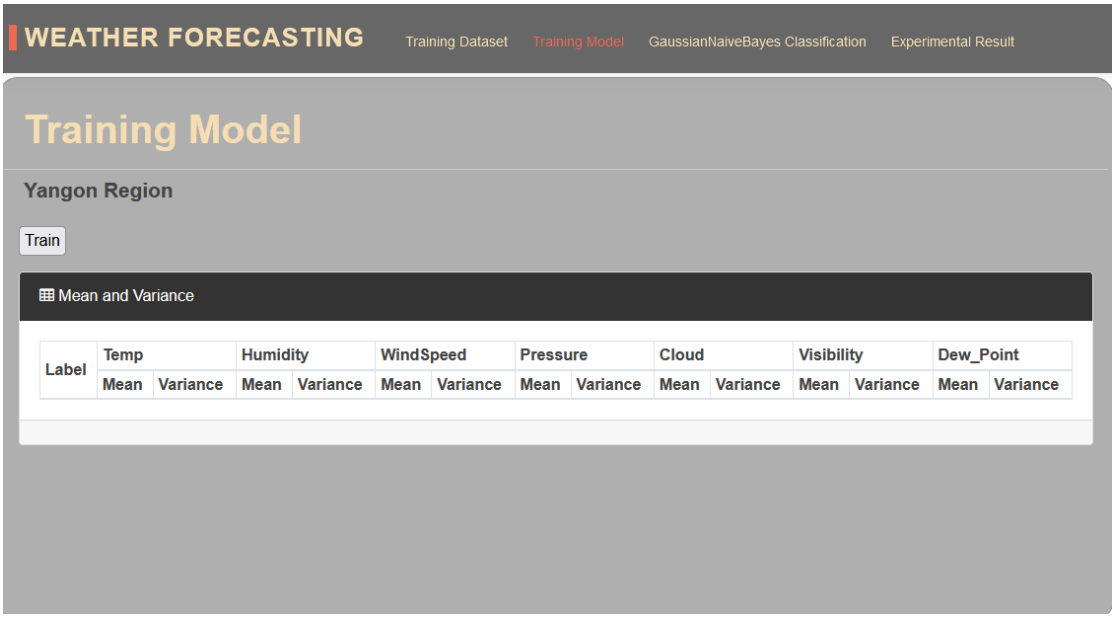


Figure 3.6 Home Page of Web Application

The "Train" button is used for training the weather dataset before the classification process. The user can also know training time as shown in Figure 3.7.

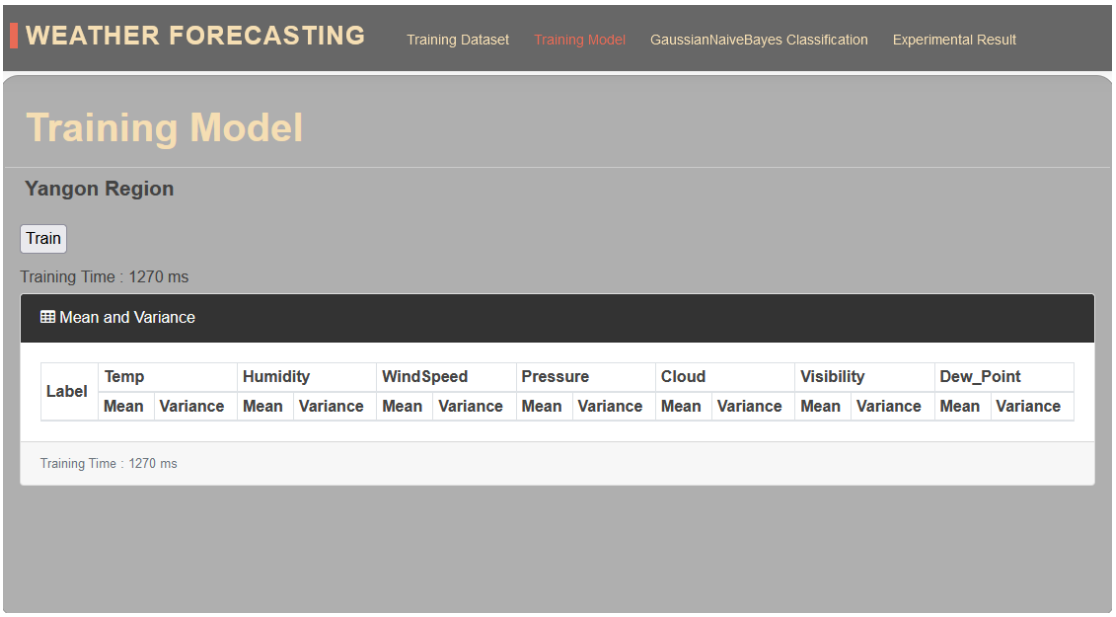
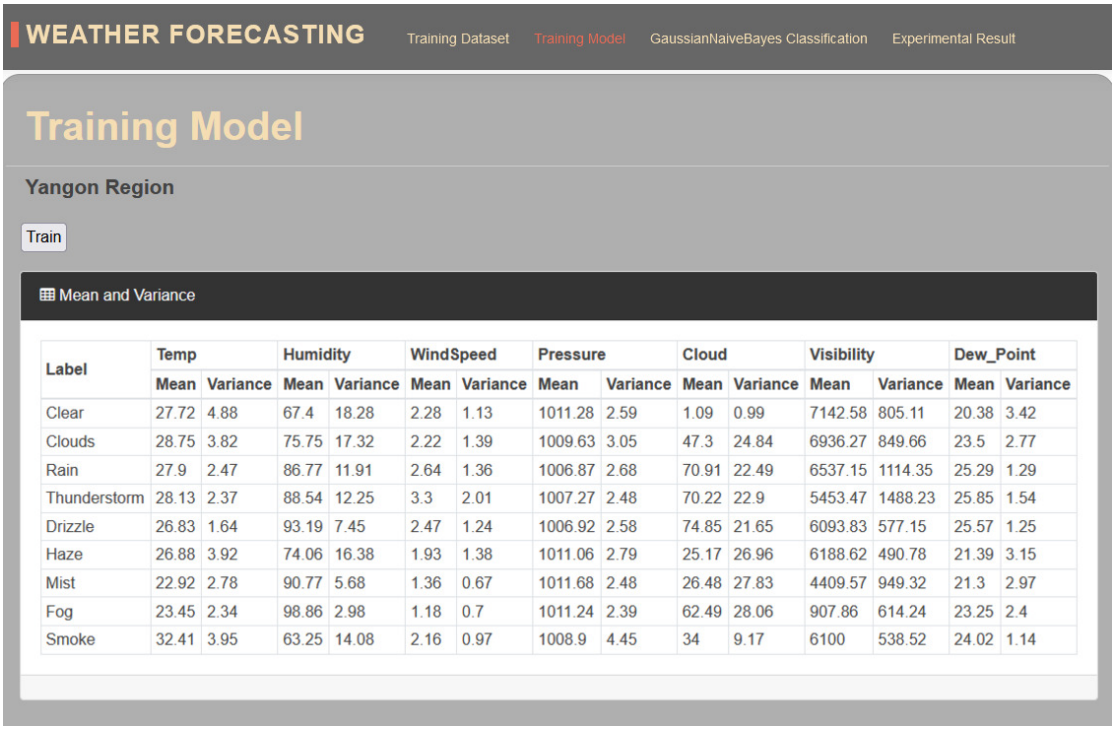


Figure 3.7 Training Time

After the training process is done, mean and variances of weather parameters calculated by Gaussian Naïve Bayes is displayed with table as shown in Figure 3.8.



Label	Temp		Humidity		WindSpeed		Pressure		Cloud		Visibility		Dew_Point	
	Mean	Variance	Mean	Variance	Mean	Variance	Mean	Variance	Mean	Variance	Mean	Variance	Mean	Variance
Clear	27.72	4.88	67.4	18.28	2.28	1.13	1011.28	2.59	1.09	0.99	7142.58	805.11	20.38	3.42
Clouds	28.75	3.82	75.75	17.32	2.22	1.39	1009.63	3.05	47.3	24.84	6936.27	849.66	23.5	2.77
Rain	27.9	2.47	86.77	11.91	2.64	1.36	1006.87	2.68	70.91	22.49	6537.15	1114.35	25.29	1.29
Thunderstorm	28.13	2.37	88.54	12.25	3.3	2.01	1007.27	2.48	70.22	22.9	5453.47	1488.23	25.85	1.54
Drizzle	26.83	1.64	93.19	7.45	2.47	1.24	1006.92	2.58	74.85	21.65	6093.83	577.15	25.57	1.25
Haze	26.88	3.92	74.06	16.38	1.93	1.38	1011.06	2.79	25.17	26.96	6188.62	490.78	21.39	3.15
Mist	22.92	2.78	90.77	5.68	1.36	0.67	1011.68	2.48	26.48	27.83	4409.57	949.32	21.3	2.97
Fog	23.45	2.34	98.86	2.98	1.18	0.7	1011.24	2.39	62.49	28.06	907.86	614.24	23.25	2.4
Smoke	32.41	3.95	63.25	14.08	2.16	0.97	1008.9	4.45	34	9.17	6100	538.52	24.02	1.14

Figure 3.8 Train Model of Web Application

The user can test manually the weather prediction result by using our "Gaussian Naïve Bayes Classification" page as shown in Figure 3.9. In this page, the user inputs the values of weather parameters by selecting the dataset row in Figure 3.10. When the required data has been extend to the system, weather classification process can be performed by clicking the "Classify" button. As a result, the predicted class is shown in Figure 3.11.

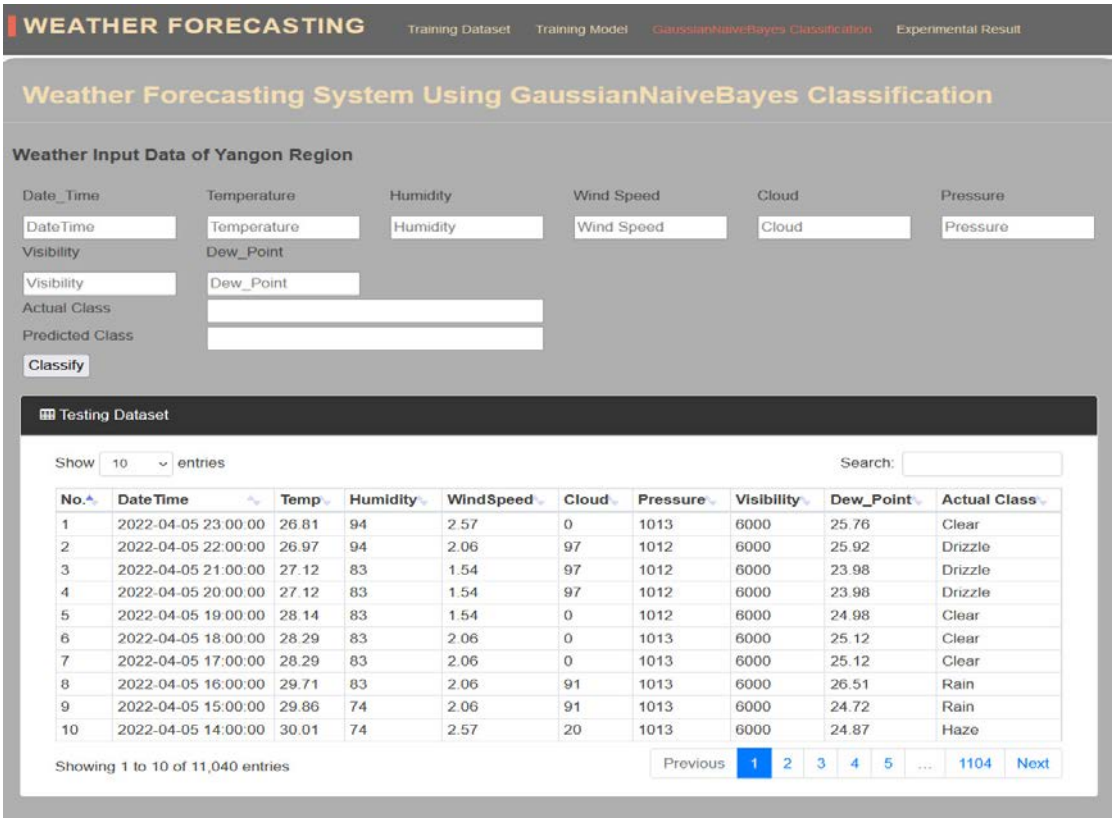


Figure 3.9 Gaussian Naïve Bayes Classification of Web Application

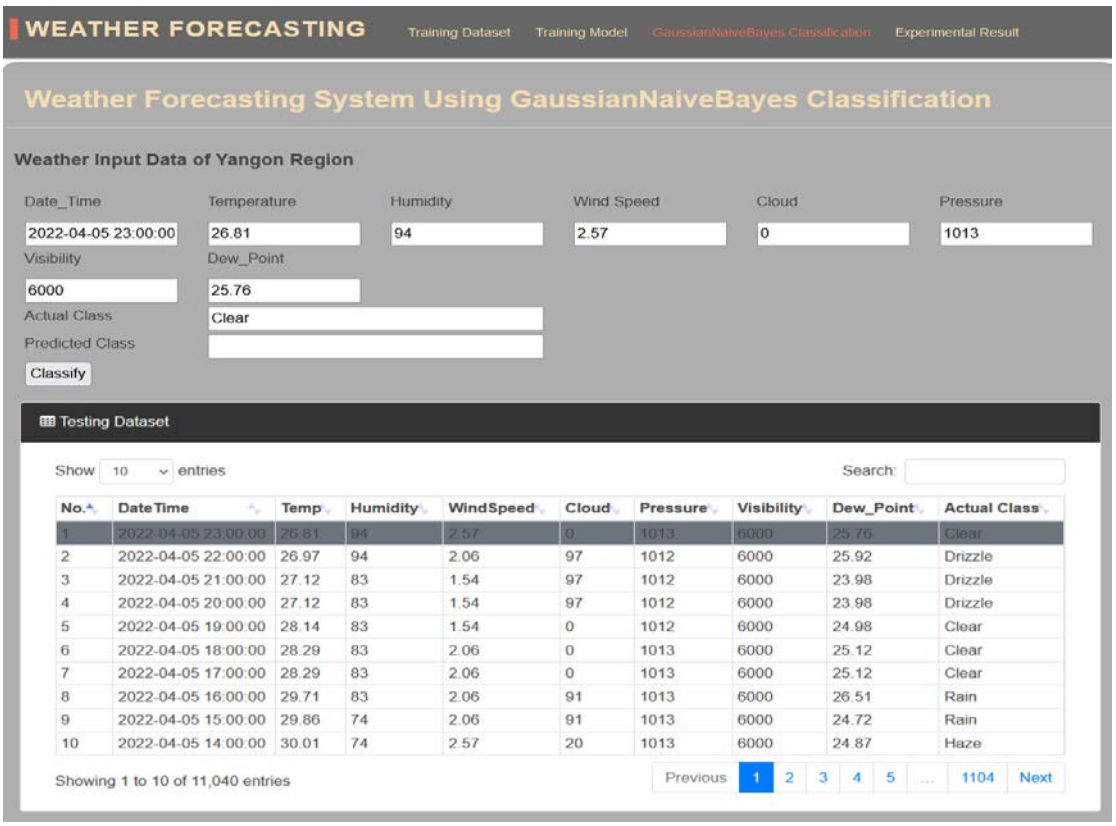


Figure 3.10 Select Data for Test

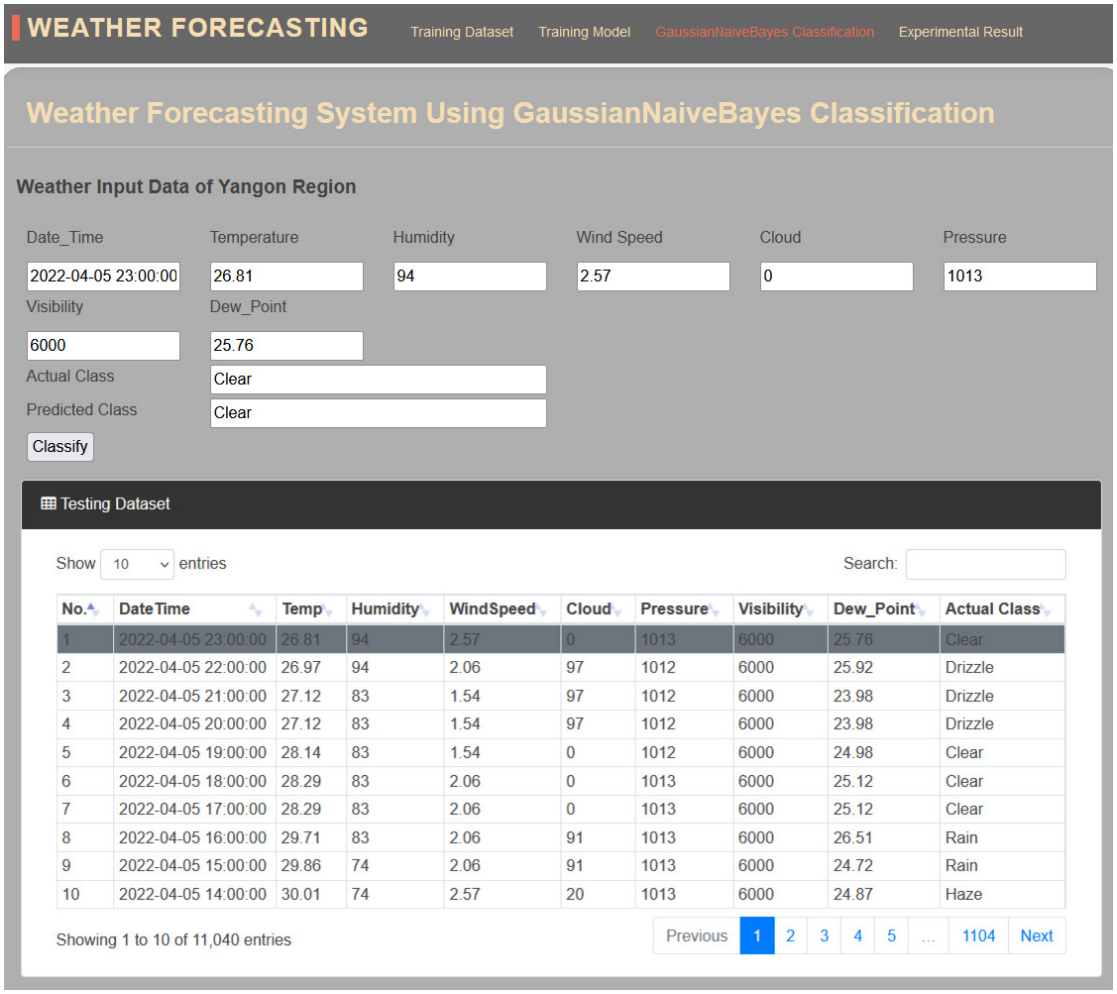


Figure 3.11 Gaussian Naïve Bayes Classification Test Page of Web Application

The “Experimental Result” page is used for displaying the confusion matrix of multiclass, and the values of precision, recall and f1-score. This page displays the accuracy of current testing, training and testing. Current testing data accuracy is shown in Figure 3.12 and Figure 3.13.

WEATHER FORECASTING Training Dataset Training Model GaussianNaiveBayes Classification Experimental Result

Experimental Result

Current Testing Data Accuracy Training Data Accuracy Testing Data Accuracy

Current Testing Result

Show 10 entries Search:

No.	Temp	Humidity	WindSpeed	Cloud	Pressure	Visibility	Dew_Point	Actual Class	Predicted Class
1	25.97	78	2.06	0	1014	7000	21.84	Clear	Clear
2	28.28	65	3.6	40	1014	6000	21.07	Haze	Haze
3	32.67	52	3.09	69	1010	7000	21.53	Clouds	Clouds
4	35.58	52	2.06	20	1006	8000	24.2	Clouds	Clouds
5	25.13	68	2.6	20	1017	7000	18.81	Haze	Haze
6	26.26	83	3.09	75	1008	6000	23.14	Drizzle	Drizzle
7	28.85	100	5.14	90	1005	4000	28.85	Thunderstorm	Thunderstorm
8	29.33	62	5.14	5	1009	7000	21.29	Clear	Clouds
9	27.12	83	1.54	97	1012	6000	23.98	Drizzle	Drizzle
10	29.33	62	5.14	5	1009	7000	21.29	Clear	Clouds

Showing 1 to 10 of 12 entries Previous 1 2 Next

Confusion Matrix

Actual \ Prediction	Clear	Clouds	Drizzle	Haze	Thunderstorm
Clear	3	2	0	0	0
Clouds	0	2	0	0	0
Drizzle	0	0	2	0	0
Haze	0	0	0	2	0
Thunderstorm	0	0	0	0	1

Figure 3.12 Current Testing Accuracy Result Page of Web Application

Precision/ Recal/ F1Score

Weather Class	Precision	Recall	F1-Score
Clear	0.6	1	0.75
Clouds	1	0.5	0.66666666666667
Drizzle	1	1	1
Haze	1	1	1
Thunderstorm	1	1	1

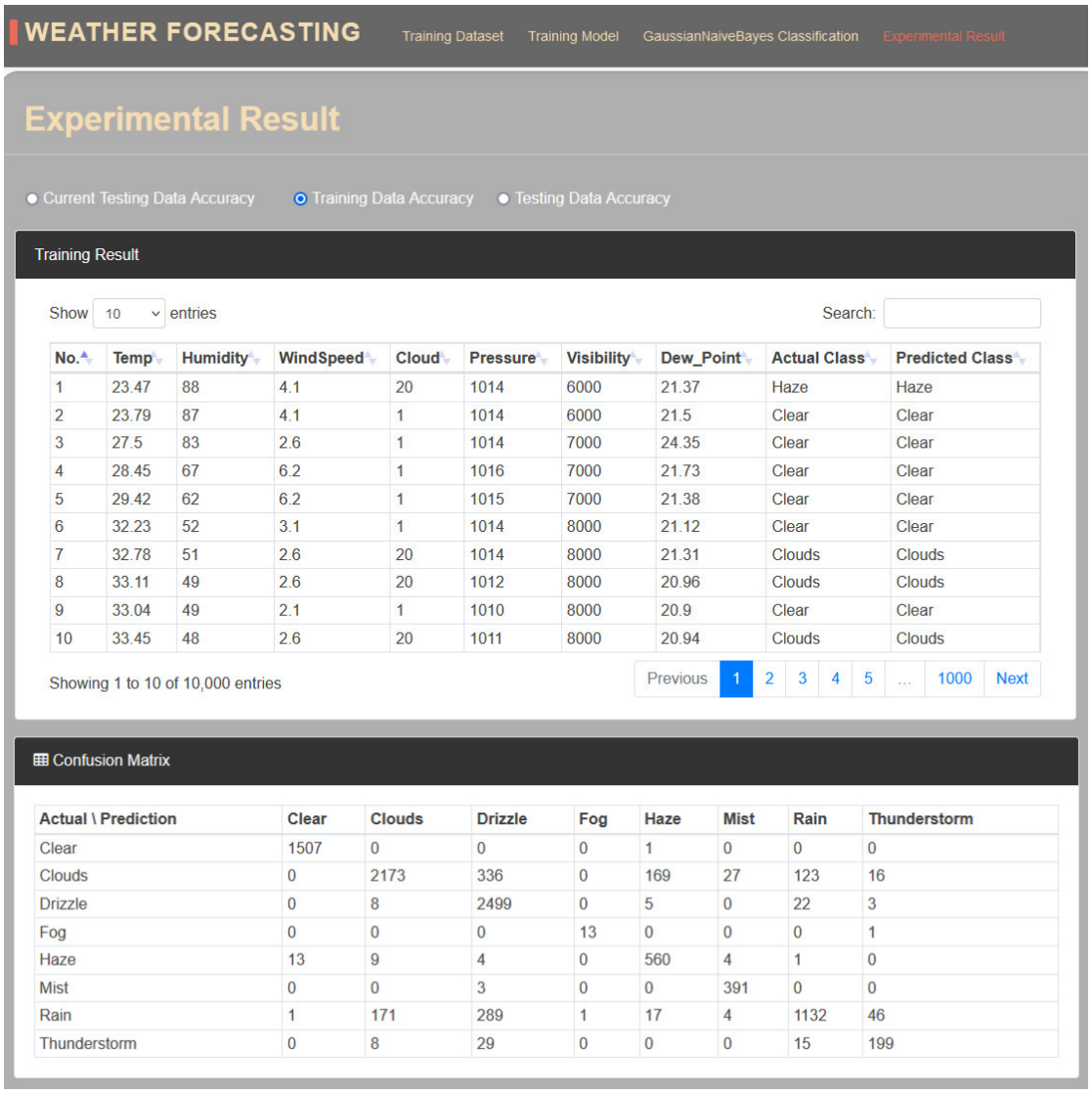
Average Precision/ Recall/ F1Score

precision	0.9
recall	0.92
f1score	0.88333333333333

Figure 3.13 Precision, Recall and F1-Score of Current Testing

After selecting “Current Testing Accuracy” radio button, the user can view result and confusion matrix table. And then the user can also view precision, recall and F1-Score depending on disarray network.

Training data accuracy is shown in Figure 3.14 and Figure 3.15.



No.	Temp	Humidity	WindSpeed	Cloud	Pressure	Visibility	Dew_Point	Actual Class	Predicted Class
1	23.47	88	4.1	20	1014	6000	21.37	Haze	Haze
2	23.79	87	4.1	1	1014	6000	21.5	Clear	Clear
3	27.5	83	2.6	1	1014	7000	24.35	Clear	Clear
4	28.45	67	6.2	1	1016	7000	21.73	Clear	Clear
5	29.42	62	6.2	1	1015	7000	21.38	Clear	Clear
6	32.23	52	3.1	1	1014	8000	21.12	Clear	Clear
7	32.78	51	2.6	20	1014	8000	21.31	Clouds	Clouds
8	33.11	49	2.6	20	1012	8000	20.96	Clouds	Clouds
9	33.04	49	2.1	1	1010	8000	20.9	Clear	Clear
10	33.45	48	2.6	20	1011	8000	20.94	Clouds	Clouds

Actual \ Prediction	Clear	Clouds	Drizzle	Fog	Haze	Mist	Rain	Thunderstorm
Clear	1507	0	0	0	1	0	0	0
Clouds	0	2173	336	0	169	27	123	16
Drizzle	0	8	2499	0	5	0	22	3
Fog	0	0	0	13	0	0	0	1
Haze	13	9	4	0	560	4	1	0
Mist	0	0	3	0	0	391	0	0
Rain	1	171	289	1	17	4	1132	46
Thunderstorm	0	8	29	0	0	0	15	199

Figure 3.14 Training Accuracy Result Page of Web Application

Precision/ Recal/ F1Score

Weather Class	Precision	Recall	F1-Score
Clear	0.99933687002653	0.99079552925707	0.99504787058435
Clouds	0.764064697609	0.91726466863656	0.83368501822367
Drizzle	0.9850216791486	0.79082278481013	0.87730384412849
Fog	0.92857142857143	0.92857142857143	0.92857142857143
Haze	0.94754653130288	0.74468085106383	0.83395383469844
Mist	0.99238578680203	0.91784037558685	0.95365853658537
Rain	0.68151715833835	0.87548337200309	0.76641841570752
Thunderstorm	0.79282868525896	0.75094339622642	0.77131782945736

Average Precision/ Recall/ F1Score

precision	0.86455030076942
recall	0.88640910463222
f1score	0.86999459724458

Figure 3.15 Precision, Recall and F1-Score of Training

After selecting “Training Accuracy” radio button, the user can view result and confusion matrix table. And then the user can also view precision, recall and F1-Score depending on disarray network.

By clicking “Testing Accuracy” radio button, the user can view result, confusion matrix table and precision, recall and F1-Score depending on disarray network. This is shown in Figure 3.16 and Figure 3.17. The detail of these results will be discussed in next chapter.

WEATHER FORECASTING Training Dataset Training Model GaussianNaiveBayes Classification Experimental Result

Experimental Result

Current Testing Data Accuracy Training Data Accuracy Testing Data Accuracy

Testing Result

Show 10 entries Search:

No.	Temp	Humidity	WindSpeed	Cloud	Pressure	Visibility	Dew_Point	Actual Class	Predicted Class
1	21.73	94	1.03	20	1015	6000	20.72	Mist	Mist
2	22.01	83	1.03	20	1015	6000	18.99	Mist	Mist
3	25.19	73	4.63	20	1016	6000	20.01	Haze	Haze
4	25.52	73	4.63	20	1016	6000	20.32	Haze	Haze
5	26.88	58	2.57	20	1016	7000	17.92	Haze	Haze
6	31.06	58	4.12	20	1015	7000	21.82	Clouds	Clouds
7	31.34	58	1.54	19	1014	8000	22.09	Clouds	Clouds
8	31.25	58	2.06	20	1013	8000	22	Clouds	Clouds
9	32.62	48	5.14	20	1012	8000	20.18	Clouds	Clouds
10	32.77	49	3.09	40	1011	8000	20.65	Clouds	Clouds

Showing 1 to 10 of 10,000 entries Previous 1 2 3 4 5 ... 1000 Next

Confusion Matrix

Actual \ Prediction	Clear	Clouds	Drizzle	Fog	Haze	Mist	Rain	Thunderstorm
Clear	2757	12	0	0	3	3	0	0
Clouds	3	2034	337	1	204	61	212	14
Drizzle	0	1	903	0	3	0	11	4
Fog	0	0	0	15	0	1	0	0
Haze	104	14	33	0	1038	18	7	1
Mist	10	1	7	4	1	466	0	1
Rain	9	158	506	2	8	7	623	74
Thunderstorm	0	6	6	0	0	0	8	109

Figure 3.16 Testing Accuracy Result Page of Web Application

Precision/ Recal/ F1Score

Weather Class	Precision	Recall	F1-Score
Clear	0.99933687002653	0.99079552925707	0.99504787058435
Clouds	0.764064697609	0.91726466863656	0.83368501822367
Drizzle	0.9850216791486	0.79082278481013	0.87730384412849
Fog	0.92857142857143	0.92857142857143	0.92857142857143
Haze	0.94754653130288	0.74468085106383	0.83395383469844
Mist	0.99238578680203	0.91784037558685	0.95365853658537
Rain	0.68151715833835	0.87548337200309	0.76641841570752
Thunderstorm	0.79282868525896	0.75094339622642	0.77131782945736

Average Precision/ Recall/ F1Score

precision	0.86455030076942
recall	0.88640910463222
f1score	0.86999459724458

Figure 3.17 Precision, Recall and F1-Score of Testing

3.4 Summary

The Gaussian Naïve Bayes based weather prediction system is implemented as the responsive web application based on OpenWeather APIs. The chief information of the proposed system is the Gaussian Naïve Bayes classification for weather class based on weather parameters received from API. All programming components are implemented by using PHP and jQuery. All functions for the proposed system can be grouped into one module: API Request and Application Module.

CHAPTER 4
EXPERIMENTAL RESULTS

In this section, the system performance is examined by using precision, recall, and f1-score of classification which depend on the multi-class disarray network. The exploratory outcomes for weather prediction are described with statistical data.

4.1 Performance Analysis

Weather prediction was tested by using historical weather dataset that include 52608 records to show the performance of the system. The training datasets are collected from the OpenWeather organization by History Bulk API.

4.1.1 Multi-Class Confusion Matrix

Confusion grid is the lattice perception of result of AI model. It is arranged in a matrix form. Confusion Matrix gives a comparison between real and anticipated values. The confusion matrix is an N x N grid, where N is the number of classes or outcomes. For 2 classes, we get 2 x 2 confusion gird. For 3 classes, we get 3 X 3 confusion gird. Confusion matrix has 4 terms to understand True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN). The representation of the confusion grid is as shown in Figure 4.1.

Actual \ Predicted Values	**POSITIVE**	**NEGATIVE**
POSITIVE	TP (True Positive)	FN (False Negative)
NEGATIVE	FP (False Positive)	TN (True Negative)

Figure 4.1 Representation of Confusion Matrix for Classification

TP, TN, FP and FN values directly as in the binary classification problem in the multi-class classification solution, These values are needed to reckon for each class and they are as follows:

- True-Positive (TP): the true positive value is where the actual value and predicted value are the same.
- False-Positive (FP): the false positive value for a class will be the sum of values of the corresponding column except for TP.
- True-Negative (TN): the true negative value for a class will be the sum of the values of all columns and rows except the values of that class that are calculating the values for.
- False-Negative(FN): the false-negative values for a class will be the sum of values of corresponding rows except for TP.

  These are shown in Figure 4.2.

Predicted				
Actual		A	B	C
	A	TP_A	E_{AB}	E_{AC}
	B	E_{BA}	TP_B	E_{BC}
	C	E_{CA}	E_{CB}	TP_C

Figure 4.2 Multi-Class of Confusion Matrix for Classification

4.1.2 Precision, Recall and F1-Score

To appraise the performance of Gaussian Naïve Bayes depend weather prediction system for weather condition, precision, recall, and F1-score methods are calculated based on multi-class confusion matrix as in Equation 4.1, 4.2, and 4.3.

Precision (P)

$$P = TP / (TP+FP) \quad 4.1$$

Recall (R)

$$R = TP / (TP+FN) \quad 4.2$$

F1-Score (F)

$$F = 2 * [(P*R) / (P+R)] \quad 4.3$$

Where, TP announces the number of class which is correctly predicted for the actual class. FP is the total number of incorrectly predicted class for all actual classes. FN denotes the total number of actual class for all incorrectly predicted.

Precision is the ability to classify the class correctly. The recall is the performance of the classification to classify all of the actual class in the dataset. This means that the precision is the exactitude and the recall is the completeness of the classification model. The f1-score is adding together into the exactitude and completeness of the system.

The multi-class confusion matrix of weather classification for Training 2000, 4000, 6000 and 10000 data from random are shown in Table 4.1, Table 4.2, Table 4.3, Table 4.4 and Table 4.5.

Table 4.1 Confusion Matrix for Training 2000 of Weather Classification

Actual \ Prediction	Clear	Clouds	Drizzle	Fog	Haze	Mist
Clear	491	1	0	0	0	0
Clouds	0	431	78	0	57	8
Drizzle	0	0	322	0	1	0
Fog	0	0	0	6	0	0
Haze	20	4	6	0	118	5
Mist	3	0	1	0	1	91

Table 4.2 Confusion Matrix for Training 4000 of Weather Classification

Actual \ Prediction	Clear	Clouds	Drizzle	Fog	Haze	Mist	Rain
Clear	1041	4	0	0	0	1	0
Clouds	0	777	128	0	95	17	57
Drizzle	0	0	674	0	1	0	1
Fog	0	0	0	7	0	0	0
Haze	30	7	6	0	282	8	2
Mist	1	0	1	0	0	139	0
Rain	3	49	152	1	6	1	396

Table 4.3 Confusion Matrix for Training 6000 of Weather Classification

Actual \ Prediction	Clear	Clouds	Drizzle	Fog	Haze	Mist	Rain
Clear	1067	0	0	0	0	0	0
Clouds	0	1194	175	0	60	24	76
Drizzle	0	0	1269	0	0	0	4
Fog	0	0	0	9	0	0	0
Haze	22	4	18	0	267	2	2
Mist	3	0	5	0	0	144	0
Rain	4	62	336	0	0	4	995

Table 4.4 Confusion Matrix for Training 8000 of Weather Classification

Actual \ Prediction	Clear	Clouds	Drizzle	Fog	Haze	Mist	Rain
Clear	1093	1	0	0	0	0	0
Clouds	0	1420	261	0	78	30	105
Drizzle	0	1	2122	0	0	0	7
Fog	0	0	0	11	0	0	0
Haze	22	1	18	0	405	3	7
Mist	1	0	8	0	0	181	0
Rain	1	86	476	1	6	9	1283

Table 4.5 Confusion Matrix for Training 10000 of Weather Classification

Actual \ Prediction	Clear	Clouds	Drizzle	Fog	Haze	Mist	Rain
Clear	1505	0	0	0	0	0	0
Clouds	0	2258	305	0	143	15	109
Drizzle	0	8	2667	0	5	0	20
Fog	0	0	0	13	0	0	0
Haze	6	7	4	0	604	4	1
Mist	0	0	2	0	0	406	0
Rain	0	155	240	0	17	2	1191

The precision, recall, and f1-score values are reckoned by equations 4.1, 4.2 and 4.3 based on above multi-class confusion matrix, and the calculation results for the weather conditions are obtained as shown in Table 4.6.

Table 4.6 Precision, Recall, and F1-Score of Training

Number of Data	Precision	Recall	F1-Score
2000	0.80794586	0.861899	0.82424
4000	0.83170943	0.875011	0.844527
6000	0.85713577	0.894201	0.866447
8000	0.85685608	0.897343	0.868039
10000	0.89548122	0.900088	0.893722

The accuracy of the performance result of training can also be seen in Figure 4.3.

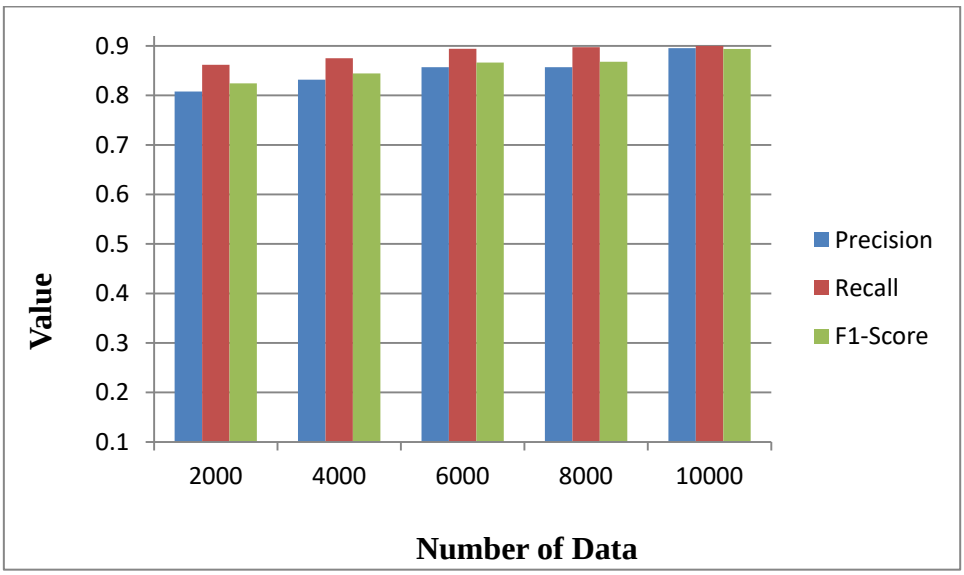


Figure 4.3 Precision, Recall, and F1-Score of Training

The multi-class confusion matrix of weather classification for Testing 2000, Testing 4000, Testing 6000, Testing 8000 and Testing 10000 from random is shown in Table 4.7, Table 4.8, Table 4.9, Table 4.10 and Table 4.11 respectively.

Table 4.7 Confusion Matrix for Testing 2000 of Weather Classification

Actual \ Prediction	**Clear**	**Clouds**	**Drizzle**	**Fog**	**Haze**	**Mist**
Clear	399	0	0	0	0	0
Clouds	0	413	75	0	40	8
Drizzle	0	1	349	0	0	0
Fog	0	0	0	7	0	0
Haze	30	7	2	0	125	3
Mist	1	0	4	0	0	72

Table 4.8 Confusion Matrix for Testing 4000 of Weather Classification

Actual \ Prediction	Clear	Clouds	Drizzle	Fog	Haze	Mist	Rain
Clear	590	1	0	0	0	0	0
Clouds	0	659	209	0	70	22	89
Drizzle	0	0	839	0	0	0	5
Fog	0	0	0	9	0	0	0
Haze	18	3	16	0	155	2	5
Mist	1	0	3	0	0	89	0
Rain	3	80	358	1	1	3	604

Table 4.9 Confusion Matrix for Testing 6000 of Weather Classification

Actual \ Prediction	Clear	Clouds	Drizzle	Fog	Haze	Mist	Rain
Clear	827	1	0	0	0	1	0
Clouds	0	1014	266	0	96	33	118
Drizzle	0	0	1314	0	0	0	13
Fog	0	0	0	7	0	0	0
Haze	31	2	29	0	259	7	5
Mist	4	0	7	0	0	132	0
Rain	6	113	515	4	2	6	912

Table 4.10 Confusion Matrix for Testing 8000 of Weather Classification

Actual \ Prediction	Clear	Clouds	Drizzle	Fog	Haze	Mist	Rain
Clear	1976	0	0	0	1	1	0
Clouds	0	1525	364	0	153	59	169
Drizzle	0	2	1211	0	0	0	11
Fog	0	0	0	23	0	1	0
Haze	65	7	20	0	360	10	8
Mist	5	0	8	0	0	247	0
Rain	6	136	570	3	6	13	748

Table 4.11 Confusion Matrix for Testing 10000 of Weather Classification

Actual \ Prediction	Clear	Clouds	Drizzle	Fog	Haze	Mist	Rain
Clear	2876	11	0	0	2	2	0
Clouds	3	2122	303	1	189	59	196
Drizzle	0	1	928	0	3	0	11
Fog	0	0	0	16	0	1	0
Haze	95	14	31	0	1117	16	7
Mist	10	1	7	3	1	487	0
Rain	9	143	469	1	8	7	625

These results of testing are shown in Table 4.12 and Figure 4.4

Table 4.12 Precision, Recall, and F1-Score of Testing

Number of Data	Precision	Recall	F1-Score
2000	0.7237776	0.751669	0.728806
4000	0.76687536	0.844085	0.783515
6000	0.7540088	0.851466	0.780371
8000	0.76927702	0.836108	0.782564
10000	0.77155028	0.850803	0.792209

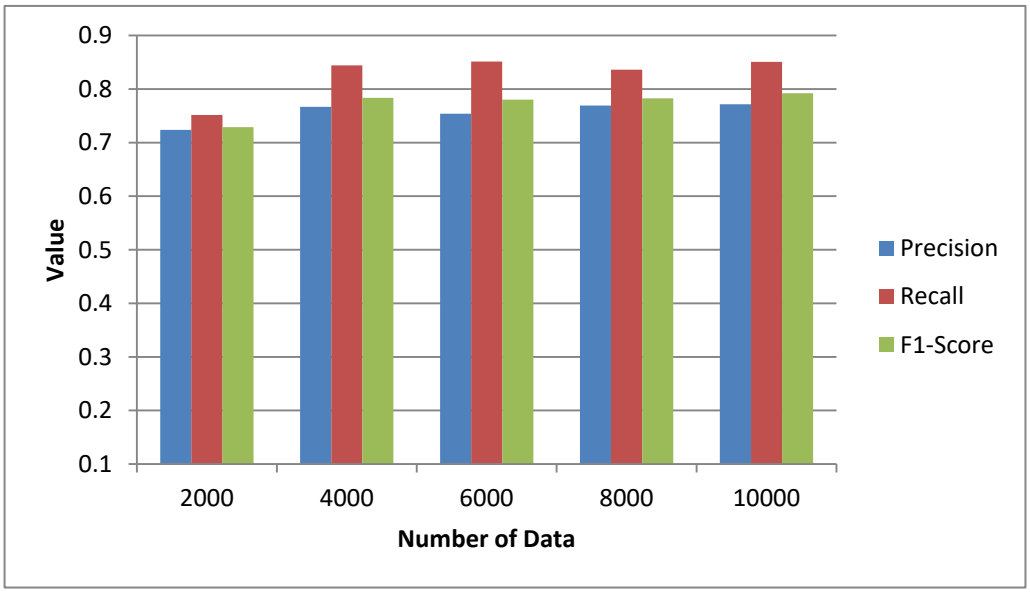


Figure 4.4 Precision, Recall, and F1-Score of Testing

The precision, recall, and f1-score values for 10 times are reckoned by equations 4.1, 4.2 and 4.3 based on above multi-class confusion matrix of training 10000, and the results for the weather conditions are obtained as shown in Table 4.13 and Figure 4.5. As 10 times for training 10000, the evaluation results prove that F1-Score is above 82%. Therefore, it has stable accuracy.

Table 4.13 Precision, Recall, and F1-Score of 10000 Training Data

Number of Test	Precision	Recall	F1-Score
1	0.876942	0.886866	0.876662
2	0.885933	0.91297	0.894792
3	0.87742	0.908011	0.88745
4	0.868714	0.902708	0.879689
5	0.863575	0.897429	0.874172
6	0.88865	0.895072	0.887313
7	0.873659	0.886633	0.874452
8	0.845879	0.865801	0.848378
9	0.830889	0.855777	0.834493
10	0.850192	0.870861	0.853628

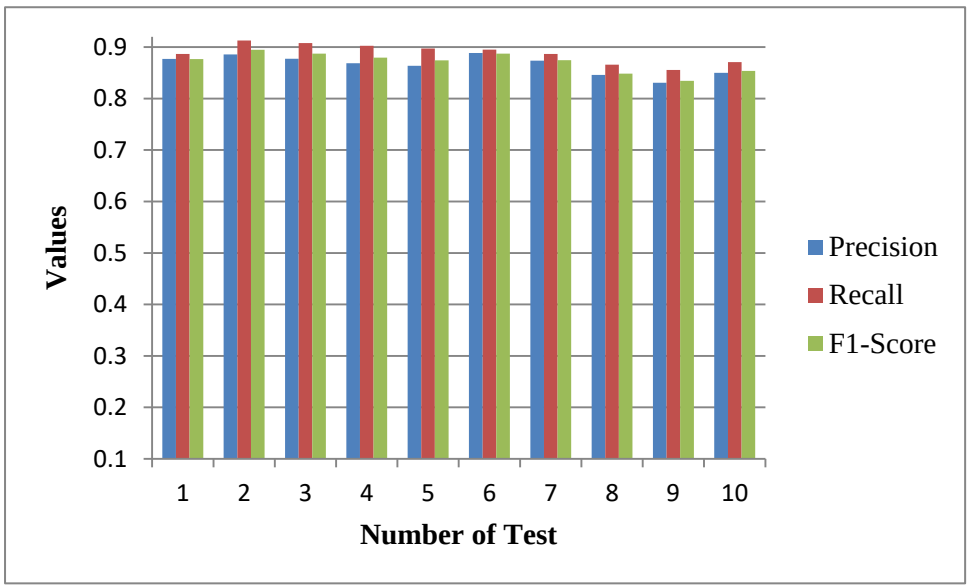


Figure 4.5 Precision, Recall, and F1-Score of 10000 Training Data

The precision, recall, and f1-score values for 10 times are also reckoned by equations 4.1, 4.2 and 4.3 based on above multi-class confusion matrix of testing 10000, and the results for the weather conditions are obtained as shown in Table 4.14

and Figure 4.6. As 10 times for testing 10000, the evaluation results prove that F1-Score is above 72% and then it has the stability of accuracy.

Table 4.13 Precision, Recall, and F1-Score of 10000 Testing Data

Number of Test	Precision	Recall	F1-Score
1	0.76686144	0.846613	0.78706135
2	0.753717	0.839837	0.77466955
3	0.74954948	0.835479	0.76967144
4	0.73465824	0.825922	0.75560689
5	0.7227215	0.818321	0.74401158
6	0.76682261	0.847888	0.78748902
7	0.76451051	0.844955	0.78454741
8	0.75753646	0.840898	0.77767968
9	0.74695837	0.837041	0.76869018
10	0.74124596	0.832727	0.76285932

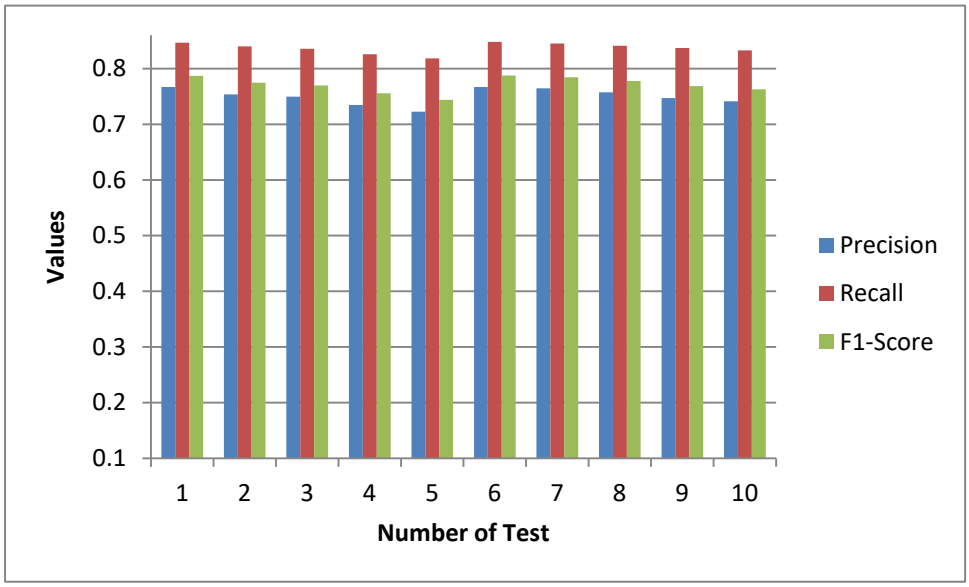


Figure 4.6 Precision, Recall, and F1-Score of 10000 Testing Data

According to the results of training, the correctness (precision) of weather prediction is obtained above 80% and the completeness (recall) is more than 85%. The average F1-Score is above 82% completely and the weather conditions can correctly be predicted. And also for testing result, the correctness (precision) of weather prediction is obtained above 72% and the completeness (recall) is more than

75%. The average F1-Score is above 72%. So, Gaussian Naïve Bayes based weather forecasting system is quite satisfactory for prediction.

4.3 Summary

In this chapter, the experimental outcomes for performance analysis of weather prediction system are described with statistical data. The analysis is performed by measuring the precision, recall, and f1-score of classification for weather prediction. According to the analysis results, the proposed Gaussian Naïve Bayes based weather prediction system is the most accurate for weather conditions. The average correctness (precision) of weather prediction is obtained above 72% and the completeness (recall) is more than 72%. The F1-Score is around 72%. The statistical data in validation results proves that the actual classes can be predicted completely by the proposed classification model.

CHAPTER 5
CONCLUSION AND FURTHER EXTENSIONS

This thesis intends to develop to forecast weather condition using Gaussian Naïve Bayes classifier which depends on seven weather parameters of historical weather dataset. The several portions of this system are found out and their contributions to the overall performance of the system are analyzed. In this chapter, the main contents of the thesis are concluded; advantages and limitations of the system, and future work are advised.

The proposed system presents the implementation of Machine Learning based weather forecasting system for weather prediction. This system introduces a system that users can use to monitor the weather conditions via any devices at anywhere at any time. The Gaussian Naïve Bayes method is used to classify the weather condition depend on seven climate information: temperature, humidity, wind speed, pressure, clouds, visibility and dew_point. Historical weather dataset plays a key role in the training process of the proposed system. Weather estimation utilizing machine learning techniques is the hard jobs because condition determining does not always mean predicting exact climate. Machine learning techniques apply few elements and foretell the climate, but the real weather conditions are dependent on multiple portions and also all those portions cannot be included in the machine learning calculation. The overall accuracy of the proposed system using the Gaussian Naïve Bayes algorithm has between 70% and 80% using training and testing. Using this proposed system can bring a lot of benefits for too many areas of agriculture.

5.1 Advantages and Limitations of the System

The proposed system serves user-friendly, high-performance, and scalable weather prediction for rainfall forecasting. As a result, the Gaussian Naïve Bayes based weather prediction is more accurate in forecasting for weather condition. The proposed web-based weather forecast application can provide monitoring weather forecast of Yangon as a meteorological station. Moreover, the use of OpenWeather History API is efficient to collect the large amount of historical weather data in a

time-saving manner. The Gaussian Naïve Bayes classifier is relatively faster in training compared to other algorithms.

However, Naïve Bayes does not have over fitting issue as it cannot represent complex features; therefore it is not reliable for datasets having many complex features. If there is a too much dependency of features on one another it will fail to give the accuracy.

5.2 Further Extensions

The proposed system is tested by using only the information for one local town. The dataset can be extended for multi towns from Myanmar. Moreover, the proposed system can be transformed to the prototype of local weather station by using the embedded multi sensors, such as Arduino Mega Board, Temperature and Dampness Detector, Anemometer, Pressure Detector and Raindrop Detector, which produces more accurate results for specific local area.

REFERENCES

[1] A. Geetha and G. M. Nasira, "Data mining for meteorological applications: Decision trees for modeling rainfall prediction", Computational Intelligence and Computing Research (ICCIC), IEEE International Conference on, 2014.

[2] A. Gustafson "Adaptive Web Design", 2nd Ed. New Riders, United Stated of America, 2015.

[3] A. McCallum, K. Nigam, “A comparison of event models for Naive Bayes text classification”, (PDF). AAAI-98 workshop on learning for text categorization. Vol. 752.

[4] AF. Pauzi, MZ. Hasan, "Development of IoT Based Weather Reporting System," in International Conference on Technology, Engineering and Sciences (ICTES), Faculty of Engineering Technology, University Malaysia Perlis (UniMAP), Malaysia, 2020.

[5] Binu Thomas, Raju G. and Sonam Wangmo, “A Modified Fuzzy C-Means Algorithm for Natural Data Exploration,” World academy of Science, Engineering and Technology 49 2009.

[6] C. Moron, J. P. Diaz, D. Ferrandez and P. Saiz, “Design, Development, and Implementation of a Weather Station Prototype for Renewable Energy System”, Journal Energies, 2234, pp. 1-13, 2018.

[7] F. Bafverfeldt, “Building an Arduino-based Weather Station and Connecting it as a Slave to a Control System”, Uppsala, Sweden, 2016.

[8] F. Joe, J. Joseph, “IoT Based Weather Monitoring System for Effective Analytics,” International Journal of Engineering and Advanced Technology (IJEAT), no. 4, pp. 311– 315, 2019.

[9] https://openweathermap.org/history

[10] KavithaRani, B., and A. Govardhan. "Effective Features and Hybrid Classifier for Rainfall Prediction", International Journal of Computational Intelligence Systems 7.5 (2014): 937-951.

[11] Liu, James NK, Bavy NL Li, and Tharam S. Dillon. "An improved naive Bayesian classifier technique coupled with a novel input solution method [rainfall prediction]." IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 31.2 (2001): 249-256.

[12] M. Tektaş, "Weather forecasting using ANFIS and ARIMA models: A case study for Istanbul", Environmental Research, Engineering and Management, vol. 51, pp. 5-10, 2010.

[13] Mark Holmstrom, Dylan Liu, Christopher Vo, “Machine Learning Applied to Weather Forecasting” Stanford University, December 15, 2016.

[14] Nallakaruppan, Kumaran, “IoT based Machine Learning Techniques for Climate Predictive Analysis,” International Journal of Recent Technology and Engineering (IJRTE), no. 5, pp. 171–175, 2019.

[15] O. M. Abrahamsen, "Machine Learning in Python for Weather Forecast based on Freely Available Weather Data," The 59th Conference in Simulation and Modelling, Metropolitan University, Norway, September 2018.

[16] P. Lynch, “The Emergence of Numerical Weather Prediction”, Cambridge University Press, 2006.

[17] Prathap N Kashyap, Preetham M.S, Rohan Venkatesh Nayak, "Smart Weather Prediction Techniques Using Machine Learning," International Research Journal of Modernization in Engineering Technology and Science, vol. 02, no. 07, pp. 2582-5208, 2020.

[18] R. Sallehuddin, et al., "Forecasting Time Series data using Hybrid Grey Relational Artificail Neural Network and Auto Regressive Integrated Moving Average", Journal of Applied Artificial Intelligence, vol. 23, 2007.

[19] R. Siddhant and D. Shaba, “Weather Forecasting Using Data Mining”, International Journal of Engineering Research in Computer Science and Engineering (IJERCSE), Vol 5, Issue 2, pp. 250-254, February 2018.

[20] S. Hay "Responsive Design Workflow", 1st ed. New Riders, United Stated of America. (2013).

[21] Santhanam, Tiruvenkadam, and A. C. Subhajini. "An efficient weather forecasting system using radial basis function neural network", Journal of Computer Science 7.7 (2011): 962.

[22] Sawaitul, Sanjay D., K. P. Wagh, and P. N. Chatur. "Classification and prediction of future weather by using back propagation algorithm-an approach." International Journal of Emerging Technology and Advanced Engineering 2.1 (2012): 110-113.

[23] Valmik B Nikam and B. B. Meshram, “Modeling Rainfall Prediction Using Data Mining Method: A Bayesian Approach”, 2013 5th International Conference on Computational Intelligence, Modelling and Simulation, 24-25 September 2013.

[24] .W. I. Bader and A. I. Hammouri, "Responsive Web Design", International Jounral of Computer Applications, Vol 150- N0. 2, 2016.

AUTHOR PUBLICATIONS

[1] May Theingi Kyaw, Ah Nge Htwe, “Weather Forecasting System Using Gaussian Naïve Bayes”, National Journal of Parallel and Soft Computing, University of Computer Studies, Yangon, 2022.