A Project report on

EARTHQUAKE DETECTION USING MACHINE LEARNING ALGORITHMS

Submitted in partial fulfillment of the requirements

for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

by

AKSHAYA G	**(194G1A0505)**
MANOJ A	**(194G1A0557)**
ARAVIND B	**(204G5A0501)**
DEEPTHI T	**(204G5A0503)**

Under the Guidance of

Mr. Nazeer Shaik **M.Tech., (Ph.D)**
Assistant Professor

Department of Computer Science & Engineering

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY
(AUTONOMOUS)
Rotarypuram Village, B K Samudram Mandal, Ananthapuramu - 515701

2022-2023

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

(AUTONOMOUS)

(Affiliated to JNTUA, Accredited by NAAC with 'A' Grade, Approved by AICTE, New Delhi & Accredited by NBA (EEE, ECE & CSE)

Rotarypuram Village, BK Samudram Mandal, Ananthapuramu-515701

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Certificate

This is to certify that the project report entitled **EARTHQUAKE DETECTION USING MACHINE LEARNING ALGORITHMS** is the bonafide work carried out by **Akshaya G, Manoj A, Aravind B, Deepthi T** bearing Roll Number **194G1A0505, 194G1A0557, 204G5A0501, 204G5A0503** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science & Engineering** during the academic year 2022-2023.

Project Guide

Mr. Nazeer Shaik M.Tech., (Ph.D)

Assistant Professor

Head of the Department

Mr. P. Veera Prakash M.Tech., (Ph.D)

Assistant Professor

Date:

Place: Rotarypuram

EXTERNAL EXAMINER

ACKNOWLEDGEMENTS

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose constant guidance and encouragement crowned our efforts with success. It is a pleasant aspect that we have now the opportunity to express our gratitude for all of them.

It is with immense pleasure that we would like to express our indebted gratitude to our Guide **Mr. Nazeer Shaik, Assistant Professor, Computer Science & Engineering**, who has guided us a lot and encouraged us in every step of the project work. We thank him for the stimulating guidance, constant encouragement and constructive criticism which have made possible to bring out this project work.

We express our deep felt gratitude to **Dr. B. Harichandana, Associate Professor,** and **Mrs. S. Sunitha, Assistant Professor, Project Coordinators** for their valuable guidance and unstinting encouragement enabled us to accoumplish our project successfully in time.

We are very much thankful to **Mr. P. Veera Prakash, Assistant Professor & Head of the Department, Computer Science & Engineering,** for his kind support and for providing necessary facilities to carry out the work.

We wish to convey our special thanks to **Dr. G. Bala Krishna, Principal** of **Srinivasa Ramanujan Institute of Technology (Autonomous)** for giving the required information in doing our project work. Not to forget, We thank all other faculty and non- teaching staff, and our friends who had directly or indirectly helped and supported us in completing our project in time.

We also express our sincere thanks to the Management for providing excellent facilities**.**

Finally, we wish to convey our gratitude to our families who fostered all the requirements and facilities that we need.

Project Associates

194G1A0505

194G1A0557

204G5A0501

204G5A0503

DECLARATION

We Ms. G. Akshaya bearing reg no : 194G1A0505, Mr. A. Manoj bearing reg no : 194G1AO557, Mr. B. Aravind bearing reg no : 204G5A0501, Ms. T. Deepthi bearing reg no: 204G5A0503 students of **SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY**, Rotarypuram, hereby declare that the dissertation entitled "**EARTHQUAKE DETECTION USING MACHINE LEARNING ALGORITHMS**" embodies the report of our project work carried out by us during IV year Bachelor of Technology under the guidance of **Mr. Nazeer Shaik**, Assistant Professor ,Department of CSE and this work has been submitted for the partial fulfillment of the requirements for the award of the Bachelor of Technology degree.

The results embodied in this project have not been submitted to any other University of Institute for the award of any Degree or Diploma.

G. AKSHAYA	Reg no: 194G1A0505
A. MANOJ	Reg no: 194G1A0557
B. ARAVIND	Reg no: 204G5A0501
T. DEEPTHI	Reg no: 204G5A0503

CONTENTS

Page No.

List of Figures

LIST OF ABBREVIATIONS

IDE	Integrated Development Environment
SVM	Support Vector Machine
RF	Random Forest
KNN	K-Nearest Neighbor
ML	Machine Learning
DT	Decision Tree

ABSTRACT

An Earthquake is the sudden shaking of the surface of the earth resulting from sudden release of energy in the Lithosphere. Reliable prediction of earthquakes has numerous societal and engineering benefits. In recent years, the exponentially rising volume of seismic data has led to the development of several automatic earthquake detection algorithms through machine learning approaches. Different algorithms have been applied on earthquake detection like Decision tree, Random Forest classifier, Support vector Machine, KNeighborsClassifier.

Earthquake detection using machine learning is a complex and evolving field that has the potential to significantly improve early warning systems and disaster response efforts. By improving the accuracy and speed of earthquake detection, machine learning can help people prepare for and respond to earthquakes more effectively.

Keywords: Machine Learning, Earthquake, Random Forest Classifier, Decision tree, Support Vector Machine, KNeighborsClassifier.

CHAPTER - 1

INTRODUCTION

Earthquake detection refers to the process of identifying and measuring seismic activity, including the vibrations and waves that are generated by the movement of tectonic plates in the Earth's crust. Earthquakes are among the most powerful and destructive natural disasters, and their detection is crucial for understanding and mitigating their impact.

The detection of earthquakes is typically carried out using a variety of tools and techniques, including seismometers, which are instruments that measure the movement of the ground, and other devices such as accelerometers, tiltmeters, and strain meters. These devices are placed in various locations throughout the Earth's crust, including on the surface, in boreholes, and on the seafloor.

When an earthquake occurs, it generates seismic waves that can be detected by these instruments, and the resulting data can be used to determine the location, magnitude, and other characteristics of the earthquake. This information is then used by scientists and emergency responders to assess the potential impact of the earthquake and to take appropriate actions to mitigate its effects.

Earthquakes are a natural disaster that can cause significant damage and loss of life. Detecting earthquakes early is crucial for reducing their impact and ensuring the safety of people living in affected areas. Traditional methods of earthquake detection rely on seismometers and other specialized equipment, which can be expensive and require significant resources to operate. Machine learning algorithms offer a promising alternative for earthquake detection. By analyzing seismic data and identifying patterns, these algorithms can help predict the likelihood of an earthquake occurring in a given area. This can provide early warning signals and help initiate disaster response efforts, potentially saving lives and reducing the impact of earthquakes. Various machine learning algorithms can be used for earthquake detection, each with its own strengths and weaknesses.

Once an earthquake is detected, the seismic data is analyzed to determine the location and magnitude of the earthquake. This information is then used to issue alerts and warnings to people in affected areas, which can help them prepare for and respond to the earthquake.

1.1 Problem Statement

The problem definition for earthquake detection using machine learning algorithms involves developing a model that can accurately and efficiently detect earthquakes from seismic data. To develop an earthquake detection model using machine learning algorithms, we need to gather large amounts of seismic data from various sources such as seismometers, accelerometers, and other seismic instruments. The data collected is then preprocessed to remove noise, correct for instrument errors, and convert the data into a suitable format for analysis. Once the data is preprocessed, it can be fed into a machine learning model to learn the patterns and trends associated with earthquake activity.

1.2 Objectives

The objective of using machine learning algorithms for earthquake detection is to improve the accuracy and speed of earthquake detection and to provide real-time alerts and warnings to help mitigate the effects of earthquakes. Machine learning algorithms can be used to analyze large amounts of seismic data and to identify patterns and trends that are difficult or impossible for humans to detect.

By using machine learning algorithms for earthquake detection is that they can process vast amounts of seismic data quickly and accurately. In addition, machine learning algorithms can identify patterns and trends in the data that may be difficult for humans to detect, especially in large and complex datasets.

CHAPTER - 2

LITERATURE SURVEY

Dinky Tulsi Nandwani et al. [1] research helps in reducing the effects of damage and destruction caused by aftershocks of earthquake. If we know the grade of buildings, we can take measures to reduce its weakness and try to strengthen it to face earthquake. The lower the grade of building, the more damage it is to be caused by earthquake. The higher the grade of building then it makes it easy for the building to survive an earthquake. The predicted output of buildings data helps in identifying safe and unsafe buildings. Using machine learning has helped us to make earthquake less painful and severe. It is a feasible option to prevent the damage to buildings and loss of human lives.

Roxane Mallouhy et al. [2] worked on Eight machines learning algorithms. They have been tested for our work to classify the major earthquake events between negative and positive. The study has been applied to a dataset collected from a center in California, which was recording inputs for 36 years. Every machine learning technique shows different results from each other. KNN, Random Forest and MLP are the best by producing the least false output (FP) while SVM, KNN, MLP and Random Forest classify the higher number of outputs correctly.

Vindhya Mudgal et al. [3] has proposed AI-based approaches have opened up new opportunities for enhancing the prediction process due to their greater precision when compared to conventional procedures. The BP-AdaBoost model, which has high percent accuracy, came out on top, followed by random forest and the support vector machine stacking model, which has low percent accuracy. Each machine learning approach yields outcomes that differ from one another.

Dr. S. Anbu Kumar et al. [4] has proposed earthquake prediction was performed, by training different Machine Learning models on seismic and acoustic data collected from a laboratory micro-earthquake simulation. During this research, six machine learning techniques including Linear Regression, Support Vector Machine, Random Forest Regression, Case Based Reasoning, XGBoost and Light Gradient Boosting Mechanism are separately applied and accuracies in the training and testing datasets were compared to pick out the best model. Light Gradient Boosting Model (LGBM)

performs well as compared to its rest competitors, it has a fair balance between Mean Absolute Error (MAE) time to failure, and range of observations, and also has the least outliers.

Alyona Galkina et al. [5] has proposed a method which aims to systematize the methods used and analyze the main trends in making predictions using machine learning. The main approaches in application of machine learning methods to a problem of earthquake prediction are observed. The main open-source earthquake catalogs and databases are described. The definition of main metrics used for performance evaluation is given.

Nakshatra et al. [6] introduces us to the idea that a strong earthquake is accompanied by means of aftershocks. s. AR picker algorithm used to determine values of P-wave and S-wave arrival time which can be treated as extracted function. Waveform is then converted into ASCII layout. Statistics is then fed to extraordinary gadget getting to know models-SVM, decision tree, Random Forest and linear regression for evaluation reason. Random forest distinguishes among earthquake leading and non-earthquake main statistics the pleasant, with an accuracy of 90.

D. G. Cortés et al. (2018) In study [7], which was published in Computers & Geosciences in 2018, an attempt to predict magnitude of the largest seismic event within the next seven days was made. The problem of earthquake prediction was treated as a regression task: four regressors (generalized linear models, gradient boosting machines, deep learning and random forest) and ensembles for them were applied. The most effective regressor was random forest (RF), yielding a mean absolute error of 0.74 on average. RF was also one of the fastest, taking only 18 minutes to train the regression models on all data. Particularly, the most accurate predictions of RF were made for moderate earthquakes (magnitudes within a range on [4, 7); MAE<=0.26. Based on these results, the authors concluded that using more complex regressor ensembles would improve the accuracy of predictions for quakes of large magnitude.

Pratiksha Bangar et al. [8] which was published in Computers & Geosciences in 2020, an accurate forecaster is designed and developed, a system that will forecast the catastrophe. It. Data-sets for Indian sub-continental along with rest of the World are collected from government sources. Pre-processing of data is followed by construction of stacking model that combines Random Forest and Support Vector Machine Algorithms. Algorithms develop this mathematical model reliant on "training data-set". Model looks for pattern that leads to catastrophe and adapt to it in its building, so as to settle on choices and forecasts without being expressly customized to play out the task. After forecast, we broadcast the message to government officials and across various platforms.

CHAPTER - 3

ANALYSIS

3.1 EXISTING SYSTEM

In the existing system Neural networks have been investigated for predicting the magnitude of the largest seismic event based on the analysis of seismicity indicators. Seismicity indicators are mathematically computed parameters that are based on earthquake data and can be used to assess the likelihood of future earthquakes.

The selection of seismicity indicators is typically based on the Gutenberg-Richter and characteristic earthquake magnitude distribution, which are empirical relationships between earthquake frequency and magnitude. Other recent earthquake prediction studies may also be used to inform the selection of seismicity indicators.

Neural networks are a type of machine learning algorithm that can be trained to learn complex relationships between inputs (in this case, seismicity indicators) and outputs (in this case, the predicted magnitude of the largest seismic event). The neural network is trained on a dataset of historical earthquake data, where the input is a set of seismicity indicators and the output is the magnitude of the largest seismic event.

3.1.1 Disadvantages

- The efficiency of existing algorithms is limited.
- Less Accuracy.
- Incorrect Predictions.

3.2 PROPOSED SYSTEM

Our proposed system for earthquake detection involves using a variety of machine learning algorithms, including the Random Forest classifier, Support Vector Machine, Decision tree, and K Nearest Neighbor, to accurately predict the level of magnitude of future earthquake occurrences. We will collect seismic data, process it, and train our models on historical earthquake data to learn the patterns and trends that are indicative of future seismic events. We will then use the trained models to predict the level of magnitude of future earthquakes and continually update and improve them as new data becomes available

- **SVM (Support Vector Machine)**

SVM uses a classifier that categorizes the info set by setting an optimal hyperplane between data. This classifier is choosen as it is incredibly versatile in the number of different kernel functions that can be applied, and this model can yield a high predictability rate. .

- **Random Forest**

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees.

- **KNN (K-Nearest Neighbor)**

K-Nearest Neighbors Algorithm. The k-nearest neighbor algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point.

- **Decision Tree**

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules, and each leaf node represents the outcome.

Overall, the decision tree algorithm can help improve the accuracy of earthquake detection by identifying the most important features for classification, allowing for the creation of more complex models that capture the nuances of the seismic data.

3.2.1 Advantages

- Increase the performance and Accuracy.
- Increase the Efficiency.

3.3 Methodology

The system uses machine learning to make predictions of the Thyroid disease and Python as the programming language since Python has been accepted widely as a language for experimenting in the machine learning area. Machine learning uses historical data and information to gain experiences and generate a trained model by training it with the data. This model then makes output predictions. The better the collection of dataset, the better will be the accuracy of the classifier. It has been observed that machine learning methods such as regression and classification perform better than various statistical models.

3.4 Machine learning

Machine Learning is undeniably one of the most influential and powerful technologies in today ‘s world. Machine learning is a tool for turning information into knowledge. In the past 50 years, there has been an explosion of data. This mass of data is useless; we analyse it and find the patterns the valuable underlying patterns within complex data that we would otherwise struggle to discover.

Basic Terminology

- **Dataset:** A set of data examples, which contain features important to solving the problem.
- **Features:** Important pieces of data that help us understand a problem.

These are fed into a Machine Learning algorithm to help it learn.

- **Model:** The representation (internal model) of a phenomenon that a Machine Learning algorithm has learnt. It learns this from the data it is shown during training. The model is the output you get after training an algorithm. For example, a decision tree algorithm would be trained and produce a decision tree model.

3.4.1 Types of Machine Learning

There are multiple forms of Machine Learning; supervised, unsupervised, semi supervised and reinforcement learning. Each form of Machine Learning has differing approaches, but they all follow the same underlying process and theory.

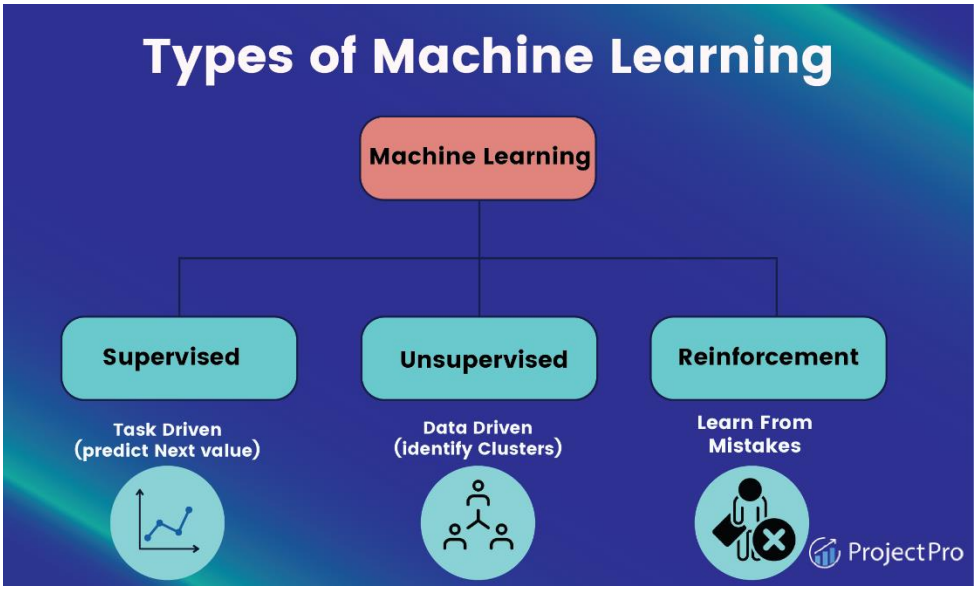


Fig 3.1: Types of Machine Learning

Supervised Learning: It is the most popular paradigm for machine learning. Given data in the form of examples with labels, we can feed a learning algorithm these example-label pairs one by one, allowing the algorithm to predict the labelfor each example, and giving it feedback as to whether it predicted the right answer or not. Over time, the algorithm will learn to approximate the exact nature of the relationship between examples and their labels. When fully- trained, the supervised learning algorithm will be able to observe a new, never before-seen example and predict a good label for it.

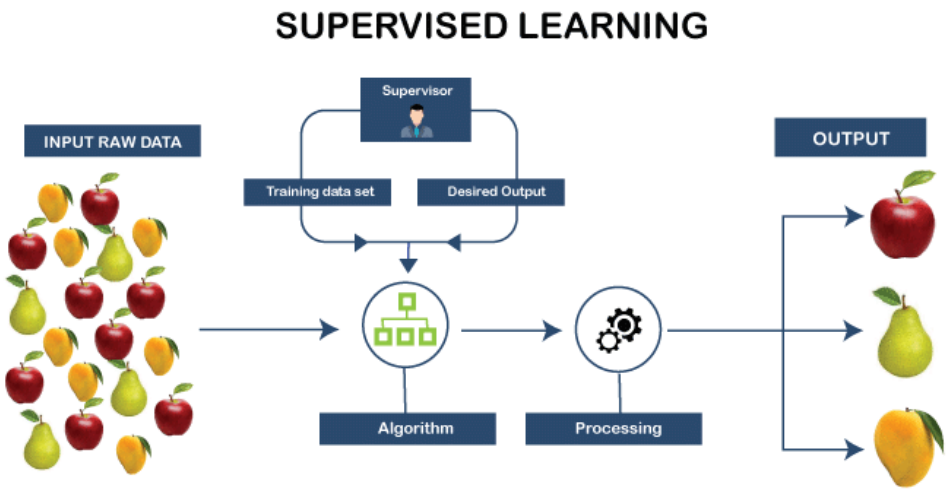


Fig 3.2: Process of Supervised Learning

Unsupervised learning: It is very much the opposite of supervised learning. It features no labels. Instead, the algorithm would be fed a lot of data and given the tools to understand the properties of the data. From there, it can learn to group, cluster, and organize the data in a way such that a human can come in and make sense of the newly organized data. Because unsupervised learning is based upon the data and its properties, we can say that unsupervised learning isdata- driven. The outcomes from an unsupervised learning task are controlled by the data and the way it‘s formatted.

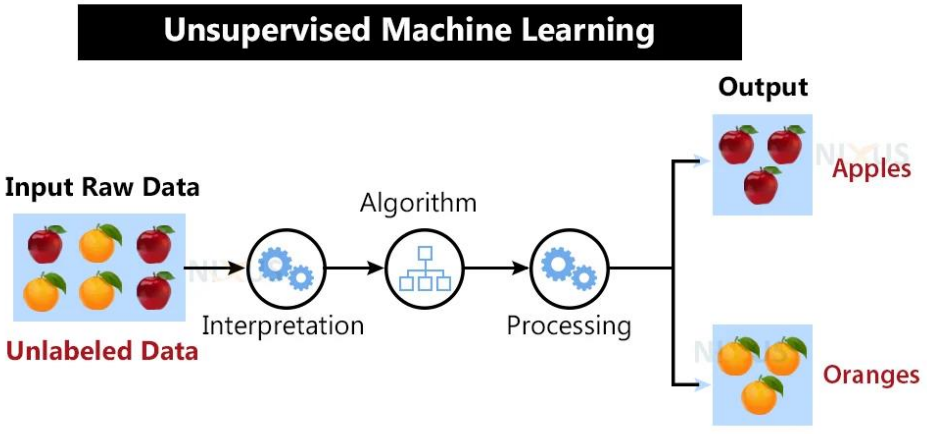


Fig 3.3: Process of Unsupervised Learning

Reinforcement learning: It is fairly different when compared to supervised and unsupervised learning. Reinforcement learning is very behaviour driven. It has influences from the fields of neuroscience and psychology. For anyreinforcement learning problem, we need an agent and an environment as wellas a way to connect the two through a feedback loop. To connect the agent to the environment, we give it a set of actions that it can take that affect the environment. To connect the environment to the agent, we have it continually issue two signals to the agent: an updated state and a reward (our reinforcementsignal for behaviour).

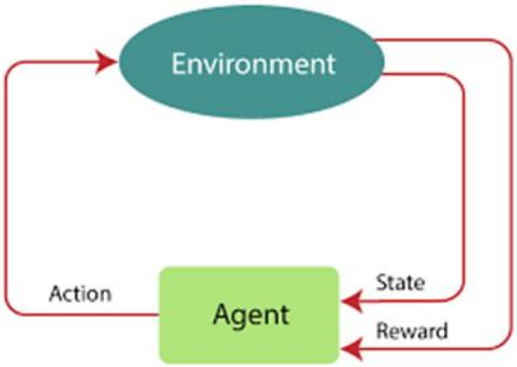


Fig 3.4: Reinforcement Learning

Machine Learning offers a wide range of algorithms to choose from. These are usually divided into classification, regression, clustering and association. Classification and regression algorithms come under supervised learning while clustering and association comes under unsupervised learning

- Classification: A classification problem is when the output variable is acategory, such as ―red‖ or ―blue‖ or ―disease‖ and ―no disease‖. Example: Decision Trees
- Regression: A regression problem is when the output variable is a real value, such as dollars or weight. Example: Linear Regression
- Clustering: A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behavior Example: k means clustering.
- Association: An association rule learning problem is where you wantto discover rules that describe large portions of your data, such as people that buy X also tend to buy Y.

Basic Process of Machine Learning

i. Data Collection**:** Collect the data that the algorithm will learn from.

ii. Data Preparation**:** Format and engineer the data into the optimal format, extracting important features and performing dimensionality reduction.

iii. Training **:** Also known as the fitting stage, this is where the Machine Learning algorithm actually learns by showing it the data that has been collected and prepared.

iv. Evaluation**:** Test the model to see how well it performs.

v. Tuning**:** Fine tune the model to maximize its performance.

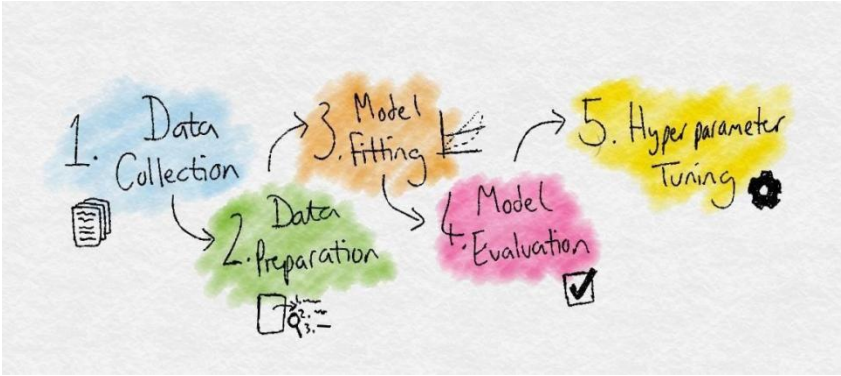


Fig 3.5: Basic process of Machine learning

3.4.2 Algorithms Used

SVM (Support Vector Machine)

SVM uses a classifier that categorizes the info set by setting an optimal hyperplane between data. This classifier is chosen as it is incredibly versatile in the number of different kernel functions that can be applied, and this model can yield a high predictability rate. Support Vector Machine is one among the foremost popular and widely used clustering algorithms. It belongs to a gaggle of generalized linear classifiers and is taken into account as an extension of the perceptron.

K-Nearest Neighbor Algorithm

K-Nearest Neighbors Algorithm. The k-nearest neighbor algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual

data point. It works by finding the distances between a query and all the examples in the data, selecting the specified number examples (K) closest to the query, then votes for the most frequent label (in the case of classification) or averages the labels (in the case of regression).

Random Forest Algorithm

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning**,** which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

Advantages

- Increase the performance
- Consistent Accuracy.
- Increase the Efficiency.

Decision Tree

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules, and each leaf node represents the outcome.

Overall, the decision tree algorithm can help improve the accuracy of earthquake detection by identifying the most important features for classification, allowing for the creation of more complex models that capture the nuances of the seismic data.

CHAPTER - 4

SYSTEM REQUIREMENTS SPECIFICATIONS

4.1 Hardware Requirements

The hardware requirements include the requirements specification of the physical computer resources for a system to work efficiently. The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. The Hardware Requirements are listed below:

1. Processor:

A processor is an integrated electronic circuit that performs the calculations that run a computer. A processor performs arithmetical, logical, input/output (I/O) and other basic instructions that are passed from an operating system (OS). Most other processes are dependent on the operations of a processor. A minimum 1 GHz processor should be used, although we would recommend S2GHz or more. A processor includes an arithmetical logic and control unit (CU), which measures capability in terms of the following:

- Ability to process instructions at a given time
- Maximum number of bits/instructions
- Relative clock speed

Fig 4.1: Processor

2. Hard Drive:

A hard drive is an electro-mechanical data storage device that uses magnetic storage to store and retrieve digital information using one or more rigid rapidly rotating disks, commonly known as platters, coated with magnetic material. The platters are

paired with magnetic heads, usually arranged on a moving actuator arm, which reads and writes data to the platter surfaces. Data is accessed in a random-access manner, meaning that individual blocks of data can be stored or retrieved in any order and not only sequentially. HDDs are a type of non-volatile storage, retaining stored data even when powered off. 50 GB or higher is recommended for the proposed system.

Fig 4.2: Hard Disk

3. Memory (RAM):

Random-access memory (RAM) is a form of computer data storage that stores data and machine code currently being used. A random- access memory device allows data items to be read or written in almost the same amount of time irrespective of the physical location of data inside the memory. In today's technology, random-access memory takes the form of integrated chips. RAM is normally associated with volatile types of memory (such as DRAM modules), where stored information is lost if power is removed, although non- volatile RAM has also been developed. A minimum of RAM is recommended for the proposed system.

Fig 4.3: RAM

4.2 Software Requirements

The software requirements are description of features and functionalities of the target system. Requirements convey the expectations of users from the software product. The requirements can be obvious or hidden, known or unknown, expected or unexpected from client's point of view.

1. Google Colab:

Google is quite aggressive in AI research. Over many years, Google developed AI framework called TensorFlow and a development tool called Colaboratory. Today TensorFlow is open-sourced and since 2017, Google made Colaboratory free for public use. Colaboratory is now known as Google Colab or simply Colab.

Another attractive feature that Google offers to the developers is the use of GPU. Colab supports GPU and it is totally free. The reasons for making it free for public could be to make its software a standard in the academics for teaching machine learning and data science. It may also have a long term perspective of building a customer base for Google Cloud APIs which are sold per-use basis.

Irrespective of the reasons, the introduction of Colab has eased the learning and development of machine learning applications.

Fig 4.5: Google Colab icon

2. Python:

It is an object-oriented, high-level programming language with integrated dynamic semantics primarily for web and app development. It is extremely attractive in the field of Rapid Application Development because it offers dynamic typing and

dynamic binding options. Python is relatively simple, so it's easy to learn since it requires a unique syntax that focuses on readability. Developers can read and translate Python code much easier than other languages. In turn, this reduces the cost of program maintenance and development because it allows teams to work collaboratively without significant language and experience barriers.

Fig 4.6: Python icon

3. VS CODE:

VS Code (Visual Studio Code) is a free and open-source source code editor developed by Microsoft. It is a popular tool used by developers for coding and debugging applications. VS Code provides features such as syntax highlighting, code completion, debugging tools, and Git integration, among others. It supports many programming languages including Java, Python, C++, and JavaScript.

One of the advantages of using VS Code is its lightweight and fast performance. It has a large number of extensions available that can enhance its functionality and make it more suitable for different kinds of development tasks. Additionally, VS Code has a customizable user interface that allows developers to configure the editor to suit their needs.

Fig 4.7: Visual studio code icon

4.3 Functional Requirements

A Functional Requirement is a description of the service that the software must offer. It describes a software system or its component. A function is nothing but inputs to the software system, its behaviour, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform. In software engineering and systems engineering, a Functional Requirement can range from the high-level abstract statement of the sender's necessity to detailed mathematical functional requirement specifications.

- Data Collection
- Data Pre-Processing
- Training and Testing
- Modelling
- Predicting

4.4 Non-Functional Requirements

Non-Functional Requirement (NFR) specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system.

Failing to meet non-functional requirements can result in systems that fail to satisfy user needs. Non-functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlogs. Example, the site should load in 3 seconds when the number of simultaneous users is > 10000. They specify the criteria that can be used to judge the operation of a system rather than specific behavior. They may relate to emergent system properties such as reliability, response time and store occupancy.

Non-functional requirements arise through the user needs, because of budget constraints, organizational policies, the need for interoperability with other software and hardware systems or because of external factors such as: - Product Requirements, Organizational Requirements, User Requirements, Basic Operational Requirement, etc.

- A requirement for ease of use; a requirement for ease of maintenance;
- The need for manageability
- The need for Security
- Requirement of availability
- Need for Scalability

4.5 Scope

- To design a model for earthquake detection with four algorithms Random Forest, decision tree, KNN and Support Vector Machine and we uses Random Forest for detection
- To detect the random forest with good accuracy.

4.6 Performance

- The performance of our project can be estimated by using the confusion matrix.
- The accuracy of this project is estimated to 97%.

CHAPTER - 5

DESIGN

5.1 System Architecture

Architecture diagrams can help system designers and developers visualize the high-level, overall structure of their system or application for the purpose of ensuring the system meets their user's needs. They can also be used to describe patterns that are used throughout the design. It's somewhat like a blueprint that can be used as a guide for the convenience of discussing, improving, and following among a team.

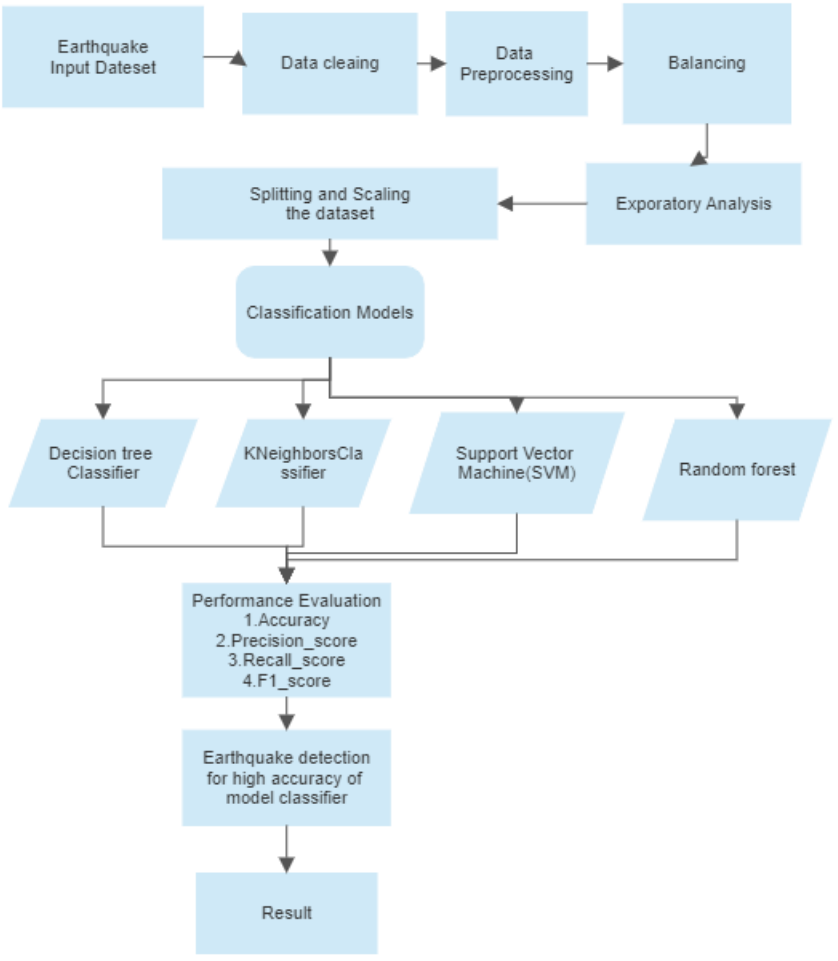


Fig 5.1: Architecture

Now, apply feature selection techniques to the rainfall dataset which has some features in it. Then a subset of features which are most important in the prediction of

floods. choose the algorithm that gives the best possible accuracy with the subset of features obtained after feature selection. Applying the algorithms to the dataset actuallymeans that it needs to train the model with the algorithms and test the data so that the model will be fit.

5.2. Steps Involved in Design

- Data Collection
- Data Preprocessing
- Model Training
- Model Evaluation

5.2.1 Data Collection

- Data is an important asset for developing any kind of Machine learning model. Data collection is the process of gathering and measuring information from different kinds of sources.
- This is an initial step that has to be performed to carry out a Machine learning project. In the present internet world these datasets are available in different websites(Ex: Kaggle, Google public datasets, Data.gov etc.)
- The dataset used in our project is downloaded from the Kaggle website and it contains nearly 116 records and 20 different attributes.
- The dataset consists of 19 independent attributes and one dependent attributes.
- So, the aim of the project is to predict the dependent variables using independent variables.

5.2.2 Data Preprocessing

- Data preprocessing is a process of preparing the raw data and making it suitable fora machine learning model. It is the first and crucial step while creating a machine learning model.
- When creating a machine learning project, it is not always the case that we come across clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put it in a formatted way. So, for this,

we use data preprocessing tasks.

- Preprocessing of the data consists of different kinds of steps in which analysis of thedata, Data cleaning, Data encoding are part of this.

5.2.2.1 Exploratory Data Analysis

- Data preprocessing is a process of preparing the raw data and making it suitable fora machine learning model. It is the first and crucial step while creating a machine learning model.
- When creating a machine learning project, it is not always the case that we come across clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put it in a formatted way. So, for this, we use data preprocessing tasks.
- Preprocessing of the data consists of different kinds of steps in which analysis of thedata, Data cleaning, Data encoding are part of this.

- ➢ Dimension reduction techniques which help create graphical display of high dimensional data containing many variables.
- ➢ Univariate visualization of each field in the raw dataset, with summary statistics.
- ➢ Bivariate visualizations and summary statistics that allows you to assess the relationship between each variable in the dataset and the target variable in the dataset and the target variable you're looking at.
- ➢ Multivariate visualizations, for mapping and understanding interactions betweendifferent fields in the data.
- ➢ This data analysis is of two types:
    a. Univariate analysis
    b. Bivariate analysis

Univariate analysis is the simplest form of data analysis where the data being analyzed contains only one variable. Since it's a single variable it doesn't deal with causes or relationships.

Bivariate data is data that involves two different variables whose values can change.Bivariate data deals with relationships between these two variables.

5.2.2.2 Filling Missing Data & Data Encoding

- ➢ The next step of data preprocessing is to handle missing data in the datasets. If our dataset contains some missing data, then it may create a huge problem for our machine learning model. Hence it is necessary to handle missing values present in the dataset.
- ➢ By calculating the mean and Mode: In this way, we will calculate the mean or Modeof that column or row which contains any missing value and will put it on the placeof missing value. This strategy is useful for the features which have numeric data such as age, salary, year, etc.
- ➢ Data encoding: Since the machine learning model completely works on mathematicsand numbers, but if our dataset would have a categorical variable, then it may createtrouble while building the model. So, it is necessary to encode these categorical variables into numbers.
- ➢ Our dataset also consists of different categorical data in which they are encoded in this step.

5.2.3 Training the Model

In this step the model is trained using the algorithms that are suitable. Flood prediction is a kind of problem in which One variable has to be determined using someindependent variables Regression model is suitable for this kind of scenario.

- A training model is a dataset that is used to train an ML algorithm. It consists ofthe sample output data and the corresponding sets of input data that have an influence on the output. The training model is used to run the input data throughthe algorithm to correlate the processed output against the sample output.
- Our project implements these algorithms like Logistic Regression, K Nearest Neighbor, Support Vector Machine, Random Forest.

5.2.4 Model Evaluation

In this step the trained model is evaluated by determining the accuracy of the model against the test data. various ways to check the performance of our machine learning or deep learning model and why to use one in place of the other. We will discuss terms like:

- Accuracy
- Recall score
- Precision score
- F1 score

Out of these we used Accuracy for evaluating our model. Accuracy is the most commonly used metric to judge a model and is actually not a clear indicator of the performance. The worst happens when classes are imbalanced balanced datasets. In such cases, other evaluation metrics such as precision, recall, and F1 score can provide a more informative picture of the model's performance.

It is important to carefully choose the evaluation metrics based on the characteristics of the dataset and the problem being solved. In the case of imbalanced datasets, using accuracy alone can lead to inaccurate conclusions about the performance of a machine learning model, and it is important to consider alternative metrics such as precision, recall, and F1 score.

5.3 UML Introduction

The unified modeling language allows the software engineer to express an analysis model using the modeling notation that is governed by a set of syntactic, semantic and pragmatic rules. A UML system is represented using five different viewsthat describe the system from a distinctly different perspective.

UML is specifically constructed through two different domains, they are:

- UML Analysis modeling, this focuses on the user model and structural model views of the systems.
- UML Design modeling, which focuses on the behavioral modeling, implementation modeling and environmental model views.

5.3.1 Usage of UML in Project

As the strategic value of software increases for many companies, the industry looks for techniques to automate the production of software and to improve quality

and reduce cost and time to the market. These techniques include component technology, visual programming, patterns and frameworks. Additionally, the development for the World Wide Web, while making some things simpler, has exacerbated these architectural problems. The UML was designed to respond to these needs. Simply, systems design refers to the process of defining the architecture, components, modules, interfaces and data for a system to satisfy specified requirements which can be done easily through UML diagrams.

5.4 Use Case Diagram:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor.

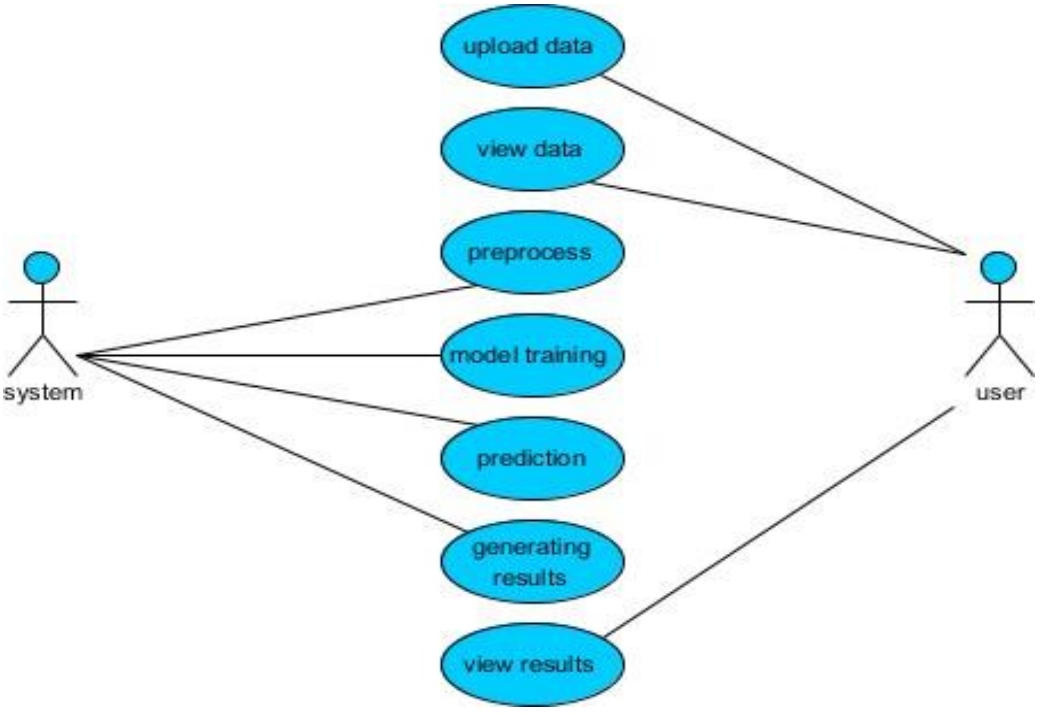
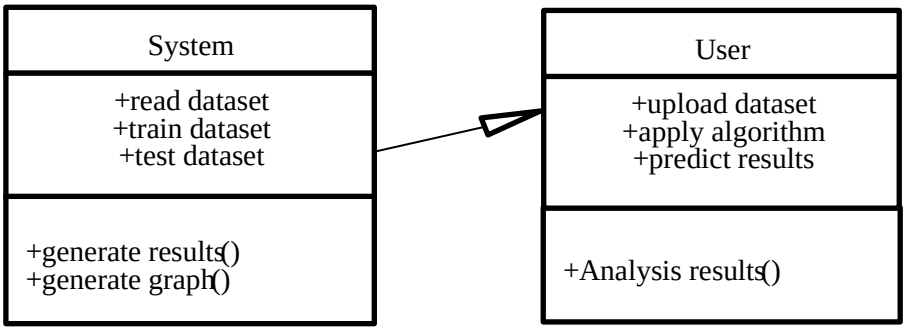


Fig 5.2: Use Case Diagram

5.5 Class Diagram:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the

relationships among the classes. It explains which class contains information.



.

Fig 5.3: Class Diagram

5.6 ACTIVITY DIAGRAM:

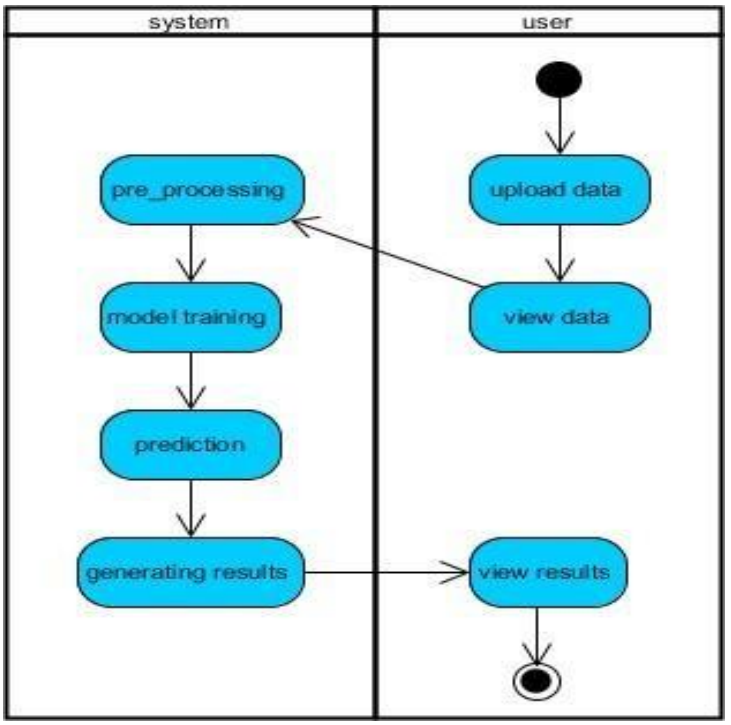


Fig 5.4: Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

CHAPTER - 6
IMPLEMENTATION

Implementation part is made using CSV file containing 20 different attributes with nearly 700 records. Mortality of earthquake detection using the data collected with Machine Learning algorithms like Logistic Decision Tree, KNN, SVM, Random Forest All these algorithms help to predict the Mortality. It is predicted by implementing all these four algorithms separately and are compared one with another.

6.1 Libraries

Python is increasingly being used as a scientific language. Matrix and vectormanipulation are extremely important for scientific computations. Both NumPy and Pandas have emerged to be essential libraries for any scientific computation, including machine learning, in python due to their intuitive syntax and high-performance matrix computation capabilities.

Pip:

The pip command is a tool for installing and managing Python packages, such as those found in the Python Package Index. It's a replacement for easy installation. Theeasiest way to install the nfl* python modules and keep them up-to-date is with a Python-based package manager called pip.

pip install (module name)

NumPy:

NumPy stands for 'Numerical Python' or 'Numeric Python'. It is an open-sourcemodule of Python which provides fast mathematical computation on arrays and matrices. Since arrays and matrices are an essential part of the Machine Learning ecosystem, NumPy along with Machine Learning modules like Scikit-learn, Pandas, Matplotlib, TensorFlow, etc. complete the Python Machine Learning Ecosystem. NumPy provides the essential multi-dimensional array-oriented computing functionalities designed for high-level mathematical functions and scientific computation. NumPy can be imported into the notebook using

import numpy as np

Pandas:

Similar to NumPy, Pandas is one of the most widely used python libraries in datascience. It provides high-performance, easy to use structures and data analysis tools. Pandas provides an in-memory 2d table object called Data frame. It is like a spreadsheetwith column names and row labels. Hence, with 2d tables, pandas are capable of providing many additional functionalities like creating pivot tables, computing columns based on other columns and plotting graphs. Pandas can be imported into Python using:

import pandas as pd.

Matplotlib:

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy One of the greatest benefits of visualization isthat it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc. Matplotlib comes with a wide variety of plots. Plots help to understand trends, patterns, and to make correlations. They're typically instruments for reasoning about quantitative information.Matplotlib can be imported into Python using:

import matplotlib. pyplot as plt

Seaborn:

Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas' data structures in Python. Visualization is the central part of Seaborn which helps in exploration and understanding of data. Seaborn offers the functionalities like Dataset oriented API to determine the relationship between variables, Automatic estimation and plotting of linear regression plots, it supports high-level abstractions for multi-plot grids and Visualizing univariate and bivariate distribution.

Flask:

Flask is a web framework for Python that enables developers to build web applications quickly and easily. It is a lightweight framework that provides only the essentials, allowing developers to add additional libraries as needed. Flask is known for its

simplicity, flexibility, and scalability, making it a popular choice for building web applications, including machine learning web apps.

Flask includes a built-in development server, enabling developers to test their web app locally before deploying it to a live server. Flask also supports various third-party libraries and extensions, including database integration, user authentication.

from flask import Flask, render_template, request

The Flask module is a web framework for Python that enables developers to build web applications quickly and easily. The render_template function allows developers to render HTML templates, while the request object enables access to incoming request data, such as form data or query parameters.

Sklearn:

Scikit-learn is a free software machine library for Python programming language. It features various classification, regression and clustering algorithms. In ourproject we have used different features of sklearn library like:

from sklearn.preprocessing import LabelEncoder

In machine learning, we usually deal with datasets which contain multiple labels in one or more than one column. These labels can be in the form of words or numbers. To make the data understandable or in human readable form, the training datais often labeled in words.

Label Encoding refers to converting the labels into numeric form so as to convertit into the machine-readable form. Machine learning algorithms can then decide in a better way on how those labels must be operated. It is an important preprocessing step for the structured dataset in supervised learning.

Label encoding converts the data in machine readable form, but it assigns a unique number (starting from 0) to each class of data. This may lead to the generation of priority issues in training of data sets..

from sklearn.preprocessing import PolynomialFeatures

Polynomial features of sklearn are mainly useful for the implementation of Polynomial regression algorithm of different kind of degrees like 1,2,3,4… This feature of sklearn help to fit the dataset with Polynomial regression algorithm. So that ithelps to Predict the sales.

from sklearn.model_selection import train_test_split

The train-test split procedure is used to estimate the performance of machine learning algorithms when they are used to make predictions on data not used to train themodel. It is a fast and easy procedure to perform, the results of which allow you to compare the performance of machine learning algorithms for your predictive modelingproblem. Although simple to use and interpret, there are times when the procedure should not be used, such as when you have a small dataset and situations where additional configuration is required, such as when it is used for classification and the dataset is not balanced.

- **Train Dataset**: Used to fit the machine learning model.
- **Test Dataset**: Used to evaluate the fit machine learning model.

Fig.6.1 Training and test data set

CSV file

The dataset used in this project is a .CSV file. In computing, a comma-separatedvalues (CSV) file is a delimited text file that uses a comma to separate values. A CSV file stores tabular data (numbers and text) in plain text. Each line of the file is a data record. Each record consists of one or more fields, separated by commas. The use of thecomma as a field separator is the source of the name for this file format. CSV is a simplefile format used to store tabular data, such as a spreadsheet or database. Files in the CSVformat can be imported to and exported from programs that store data in tables, such asMicrosoft Excel or OpenOffice Calc. Its data fields are most often separated, ordelimited, by a comma.

A CSV is a comma-separated values file, which allows data to be saved in a tabular format. CSVs look like a garden-variety spreadsheet but with a .csv extension. CSV files can be used with most any spreadsheet program, such as Microsoft Excel orGoogle Spreadsheets.

	title	magnitude	date_time	cdi	mmi	alert	Earthquake	sig	net	nst	dmin	gap	magType	depth	latitude	longitude	location	continent	country
0	M 7.0 - 18 km SW of Malango, Solomon Islands	7.0	22-11-2022 02:03	8	7	green	1	768	us	117	0.509	17.0	mww	14.000	-9.7963	159.596	Malango, Solomon Islands	Oceania	Solomon Islands
1	M 6.9 - 204 km SW of Bengkulu, Indonesia	6.9	18-11-2022 13:37	4	4	green	0	735	us	99	2.229	34.0	mww	25.000	-4.9559	100.738	Bengkulu, Indonesia	NaN	NaN
2	M 7.0 -	7.0	12-11-2022 07:09	3	3	green	1	755	us	147	3.125	18.0	mww	579.000	-20.0508	-178.346	NaN	Oceania	Fiji
3	M 7.3 - 205 km ESE of Neiafu, Tonga	7.3	11-11-2022 10:48	5	5	green	1	833	us	149	1.865	21.0	mww	37.000	-19.2918	-172.129	Neiafu, Tonga	NaN	NaN
4	M 6.6 -	6.6	09-11-2022 10:14	0	2	green	1	670	us	131	4.998	27.0	mww	624.464	-25.5948	178.278	NaN	NaN	NaN

Fig.6.2 Dataset with 15 different attributes
Earthquake_data.csv

6.2 Implementation

6.2.1 Importing all the required Modules and Libraries

All the required libraries like SK-learn and Modules like NumPy, Pandas, Matplotlib, Seaborn are imported into the Jupyter notebook initially into the filecreated in the notebook.

After importing all the modules and libraries into the notebook, A csv file has to be loaded using Pandas into the notebook. The implementation of these will be as follows:

```
import pandas as pd
import numpy as np
np.random.seed(42)

import matplotlib.pyplot as plt
import seaborn as sns
plt.style.use('ggplot')
%matplotlib inline

import warnings
warnings.filterwarnings('ignore')

from imblearn.over_sampling import SMOTE
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

#ML Algorithms
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier, GradientBoostingClassifier

from sklearn.metrics import f1_score,accuracy_score, recall_score, precision_score
from sklearn.metrics import ConfusionMatrixDisplay
from sklearn.metrics import classification_report,confusion_matrix
from  mlxtend.plotting import plot_confusion_matrix
```

Fig.6.3 Importing of modules and libraries

6.2.2 Data Visualization

As it contains large amounts of data it is not possible to analyze with the human eye normally so the feature of Data Visualization helps to analyze the entire data. The relation between any two features can be only analyzed with the Data Visualization technique. This Visualization can be made in different forms by representing the data in the pictorial forms like graph, bar chart and many other forms.

Some Visualizations are made for the dataset that is collected for the detection of earthquake. They are as follows:

6.2.2.1 Data Cleaning

In data cleaning, checking for null values is a common step in data cleaning to ensure that the dataset is complete and ready for analysis. To check for null values in a Pandas DataFrame, you can use the isnull() method to return a Boolean DataFrame indicating which cells have missing data, and then use the sum() method to count the number of missing values in each column.

```
[11] def nullValues(df):
        total=df.isnull().sum()
        percent=df.isnull().sum()/df.isnull().count()*100
        null_df=pd.concat([total,percent],axis=1,keys=["Total","Percent"])
        null_df=null_df[null_df["Percent"]>0]
        null_df=null_df.sort_values(by="Percent",ascending=False)
        print(pd.DataFrame(null_df))
        plt.figure(figsize=(16,10))
        sns.barplot(x=null_df.index,y=null_df["Percent"],color="g")
        plt.xticks(rotation=90)
        plt.xlabel("Null_value_Column")
        plt.ylabel("Percent")

nullValues(df)

                Total    Percent
continent         576  73.657289
alert             367  46.930946
country           298  38.107417
location            5   0.639386
```

Fig 6.4: checking for null values

6.2.2.2 Balancing

Balancing in machine learning refers to the process of addressing class imbalance in a dataset. Class imbalance occurs when the number of instances in one class is significantly higher than the other classes, which can lead to biased models that perform poorly on the minority class.

```
X = df[features]
y = df[target]

X = X.loc[:,~X.columns.duplicated()]


sm = SMOTE(random_state=42)
X_res, y_res= sm.fit_resample(X, y,)

y_res.value_counts().plot(kind='bar', title='Count (target)', color=['green', 'orange', 'red', 'yellow']);
```

Fig 6.5: Balancing method

To visualize the class distribution of the target variable in the dataset. If there is a significant imbalance between the classes, such as one class having a much larger count than the others, it may be necessary to balance the dataset before training a machine learning model to avoid biased performance.

6.2.4 Exploratory Analysis

Exploratory analysis is an important step in data analysis, which involves exploring and summarizing data to gain insights and identify patterns, trends, and relationships. The goal is to understand the data better, and to generate hypotheses that can be tested in further analyses.

There are several techniques and tools that can be used for exploratory analysis, including:

- **Summary statistics:** This involves calculating measures such as mean, median, standard deviation, and correlation coefficients to summarize the data.
- **Visualization:** This involves creating graphs and charts to display the data visually, such as histograms, scatterplots, and boxplots. Visualization can help to identify outliers, trends, and patterns in the data.
- **Data cleaning:** This involves identifying and addressing issues such as missing data, outliers, and inconsistencies in the data.
- **Dimensionality reduction:** This involves reducing the number of variables in

the data, for example by using principal component analysis (PCA) or factor analysis.

- **Clustering:** This involves grouping similar data points together, based on their characteristics or attributes.

```
fig, axes = plt.subplots(1, 3, figsize=(18, 6), dpi=100)

sns.histplot(data = df1, x = 'magnitude', ax=axes[0])
axes[0].set_title("Magnitude")

sns.histplot(data = df1, x = 'nst', ax=axes[1])
axes[1].set_title("NST")

sns.histplot(data = df1, x = 'mmi', ax=axes[2])
axes[2].set_title("MMI")

Text(0.5, 1.0, 'MMI')
```

Fig 6.6: Exploratory analysis of magnitude,NST,MMI

6.2.5 Magnitude vs Depth

The relationship between magnitude and depth is an important aspect of earthquake analysis, as it can provide information about the nature of the earthquake and its potential impact. Generally, larger earthquakes tend to occur at greater depths, but there are many exceptions to this rule.

One way to explore the relationship between magnitude and depth is to create a scatter plot, with magnitude on the x-axis and depth on the y-axis. Each point on the plot represents a single earthquake event, and the position of the point indicates its magnitude and depth.

6.3 Feature Engineering

What is a feature and why do we need the engineering of it? Basically, all machine learning algorithms use some input data to create outputs. This input data comprises features, which are usually in the form of structured columns. Algorithms require features with some specific characteristics to work properly. Here, the need for feature engineering arises. I think feature engineering efforts mainly have two goals:

- Preparing the proper input dataset, compatible with the machine learning algorithmrequirements.
- Improving the performance of machine learning models.

Item Visibility is one of the features of the dataset in which maximum number of

values in that row are Zeros so they have to be normalized and are set to some value. So, mean of the entire column of data visibility is calculated and the value is set to that mean value. This make all the Zeros of the column to particular value and accuracy of the model can be made more efficient.

6.3.1 Splitting the dataset

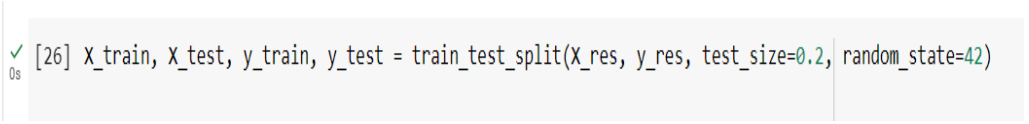


Fig 6.7: Train and Test split of data

For all the Machine learning models to train with any algorithm of their choice the dataset has to be divided into two parts called Training dataset and Testing dataset. Generally, the training dataset will be 80% of the entire dataset and 20% of the data as the Testing dataset.

Training dataset will be used to train the model and testing dataset will be used to find the accuracy of our predicted model. Performance evaluation can be made with the accuracy of the trained model with required algorithm.

Those splitting of the dataset to train and test splitting can be made using the command from the sklearn library as shown above.

x_train-Represents train dataset

x_test-Represents test dataset.

6.3.2 StandardScaler

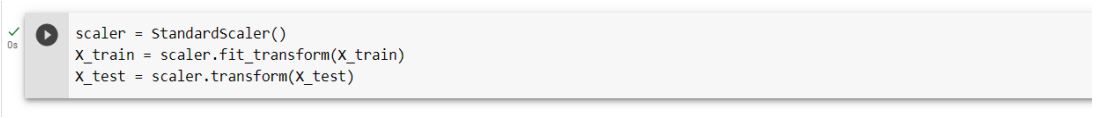


Fig 6.8: Standard Scaler

StandardScaler is a preprocessing technique in machine learning used to scale and transform the features (input variables) of a dataset. It is a widely used technique to normalize and standardize the features before training a model. This ensures that the mean of the features is 0 and the variance is 1. The formula used to calculate the standardized value of a feature 'x' is:

6.4 Model prediction

In this prediction the entire data is trained with five different models in which each model provides different output values of different accuracies. All the models are compared and the conclusions are made.

6.4.1 Modelling with KNN AlgorithmStep1: Load the dataset

Step2: Divide the dataset

Step3: Assign the KNN algorithm to a variable.

Step4: Fit the training dataset using KNN algorithm

Step5: Predict the values for the testing dataset using a trained model.

Step6: Check the accuracy of the model.

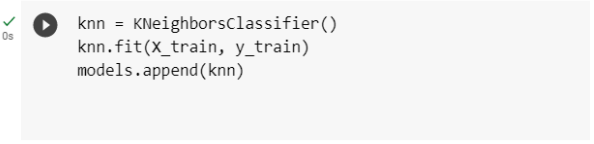


Fig 6.9: Fitting model with KNN

6.4.2 Modelling with decision Tree

Step1: Load the dataset.

Step3: Assign the Decision Tree algorithm to a variable.

Step4: Fit the training dataset using Decision Tree algorithm.

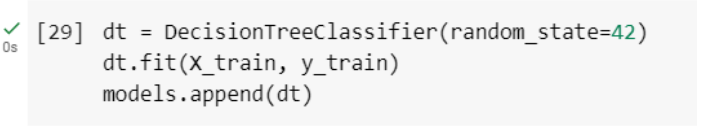


Fig 6.10: Fitting model with decision tree

6.4.3 Modelling with SVC Algorithm

Step1: Load the dataset

Step2: Divide the dataset

Step3: Assign the svc to a variable.

Step4: Fit the training dataset using svc algorithm.

Step5: Predict the values for the testing dataset using a trained model.

Step6: Check the accuracy of the model.

```
[31] svm = SVC(random_state=42)
     svm.fit(X_train, y_train)
     models.append(svm)
```

Fig 6.11: Fitting model with SVC

svc-Variable of SVC algorithm

.fit()-fit method of sklearn

.predict()-predict method for predicting the

valuesX_train,Y_train-Training data

X_test,Y_test-Testing data

6.4.4 Modelling with Random Forest Algorithm

Step1: Load the dataset

Step2: Divide the dataset

Step3: Assign the Random Forest Classifier to a variable.

Step4: Fit the training dataset using Random Forest Classifier algorithm.

Step5: Predict the values for the testing dataset using a trained model.

Step6: Check the accuracy of the model.

```
[32] rf = RandomForestClassifier(random_state=42)
     rf.fit(X_train, y_train)
     models.append(rf)
```

Fig 6.12: Fitting model with Random Forest

.fit()-fit method of sklearn

.predict()-predict method for predicting the

valuesX_train,Y_train-Training data

X_test,Y_test-Testing data

6.5 Classification metrics:

When evaluating classification models, there are several common metrics used to assess their performance. Here are some of the most common metrics and how to calculate them:

```
from sklearn.metrics import f1_score,accuracy_score, recall_score,
precision_score
```

- **Accuracy**: measures the proportion of correct predictions out of the total number of predictions. It is calculated as:

$$\text{Accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

where TP is the number of true positives (correctly predicted positive instances), TN is the number of true negatives (correctly predicted negative instances), FP is the number of false positives (incorrectly predicted positive instances), and FN is the number of false negatives (incorrectly predicted negative instances).

- **Precision score:** measures the proportion of true positive predictions out of the total number of positive predictions. It is calculated as:

$$\text{Precision} = \frac{TP}{(TP + FP)}$$

- **Recall score:** measures the proportion of true positive predictions out of the total number of actual positive instances. It is calculated as:

$$\text{Recall} = \frac{TP}{(TP + FN)}$$

- **F1 score:** is the harmonic mean of precision and recall, and provides a single score that balances the trade-off between precision and recall. It is calculated as:

$$\text{F1} = \frac{2 * (precision * recall)}{(precision + recall)}$$

6.6 Confusion matrix:

A confusion matrix is a table that is often used to evaluate the performance of a classification model. The matrix compares the predicted class of a set of data points to the actual class of those data points, and shows the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

The confusion matrix is a useful tool for evaluating the performance of a classification model because it provides a more detailed understanding of the model's performance beyond a single accuracy score. For example, it can be used to calculate metrics such as precision, recall, and F1-score, which provide insight into the model's ability to correctly identify positive cases, negative cases, or both.

6.7 Web app UI:

A web app user interface (UI) is the visual front-end that users interact with when accessing a web app. It typically includes various elements, such as buttons, forms, images, and text, that enable users to interact with the app and perform different actions.

In the context of a machine learning web app, the UI can include input fields where users can enter data to be used for prediction, as well as output fields where the predicted results are displayed. The UI can also include various other elements, such as data visualizations, charts, and graphs, that help users better understand the predicted results.

The design of the web app UI is an essential aspect of the overall user experience, as it can significantly impact how users interact with the app and their overall satisfaction with it. Therefore, developers should carefully consider the design and layout of the UI, ensuring that it is intuitive, user-friendly, and aesthetically pleasing.

CHAPTER - 7

TESTING

7.1 SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the software system meets its specified requirements and functions as expected. Testing involves creating and executing test cases to verify that the software behaves correctly under various conditions and scenarios.

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. The testing process can be manual or automated and can be conducted at various stages of the software development life cycle, such as unit testing, integration testing, system testing, and acceptance testing. Overall, testing plays a crucial role in ensuring the quality and reliability of software products.

7.2 TYPES OF TESTS

7.2.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive.

Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results. unit testing is an essential part of the software development process, which helps developers identify defects early, improve the quality of the software, and reduce the cost of bug fixes. By validating the functionality of each unit of code, developers can ensure that the software application meets its functional requirements and performs as expected.

7.2.2 Integration testing

Integration testing is a type of software testing that involves testing the interactions and interfaces between different software components that have been integrated into a larger system. The purpose of integration testing is to ensure that these components work together as expected and that the integrated system meets its functional requirements.

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

7.2.3 Functional Test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

7.3 SYSTEM TEST

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and points.

7.3.1 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

7.3.2 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works

Field testing will be performed manually and functional tests will be written in detail,

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.
- **Integration Testing**

  Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

  The task of the integration test is to check that components or software applications, e.g., components in a software system or one step up- software applications at the company level-interact without error.
- **Test Results:** All the test cases mentioned above passed successfully. No defects encountered.
- **Acceptance Testing**

  User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements. Test Results: All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER - 8

RESULTS & ANALYSIS

Evaluation of Algorithms:

Evaluates the performance of four different machine learning algorithms, including decision tree, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Random Forest.

	Name	Accuracy	Precision_score	Recall_score	F1_score
0	DecisionTreeClassifier	0.942308	0.941710	0.942308	0.941875
1	KNeighborsClassifier	0.923077	0.934314	0.923077	0.923851
2	SVM	0.930769	0.945339	0.930769	0.932321
3	RandomForestClassifier	0.976923	0.977345	0.976923	0.976927

Fig 8.1: Evaluation of algorithms

Confusion Matrix:

The model is designed to classify objects based on their color, then the x-axis of the confusion matrix would represent the predicted classes, which are the colors green, yellow, orange, and red. The y-axis would represent the actual classes of the objects. The confusion matrix would show the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) for each class.

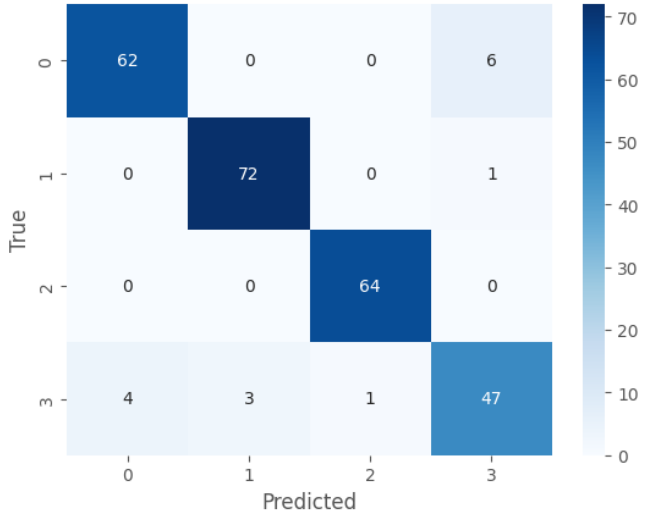


Fig 8.2: Confusion for decision tree

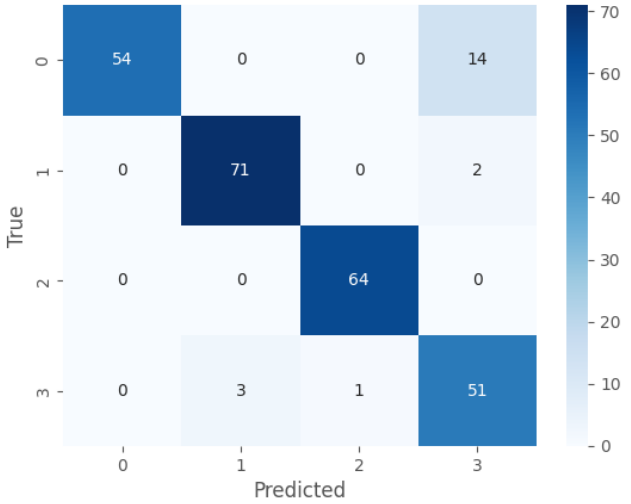


Fig 8.3: Confusion matrix for KNN

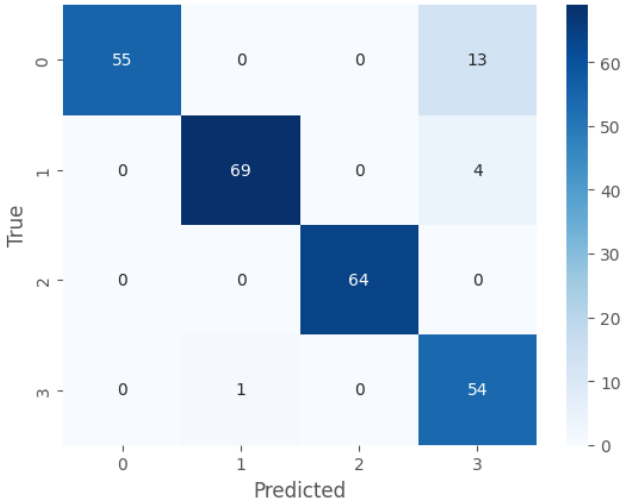


Fig 8.4: Confusion Matrix for SVM

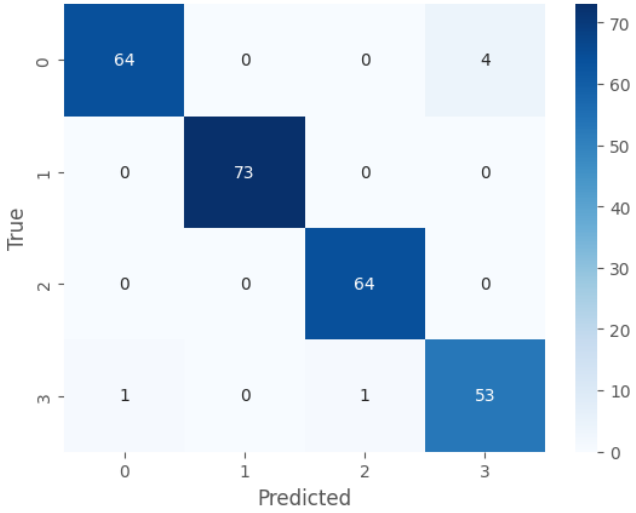


Fig 8.5: Confusion Matrix for Random Forest

Performance comparison of Model:

After evaluating the performance of the four machine learning algorithms, the results can be compared to determine which algorithm performs the best. For instance, the accuracy, precision, recall, and F1 score.

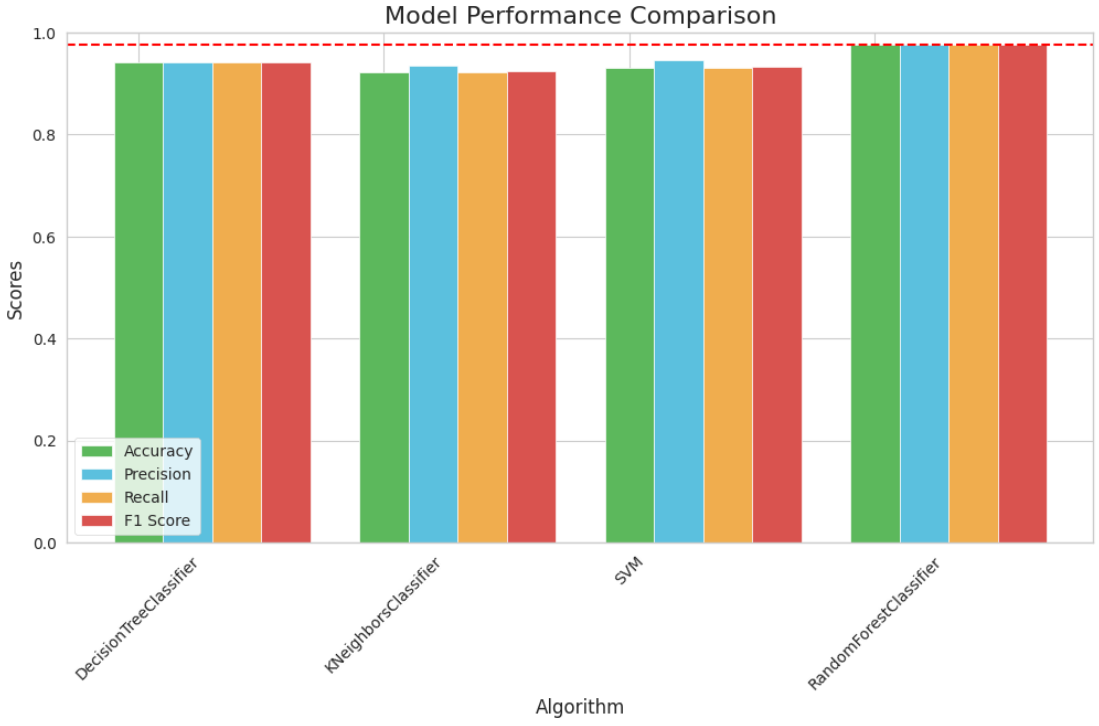


Fig 8.6: Performance comparison of algorithms

Web app results:

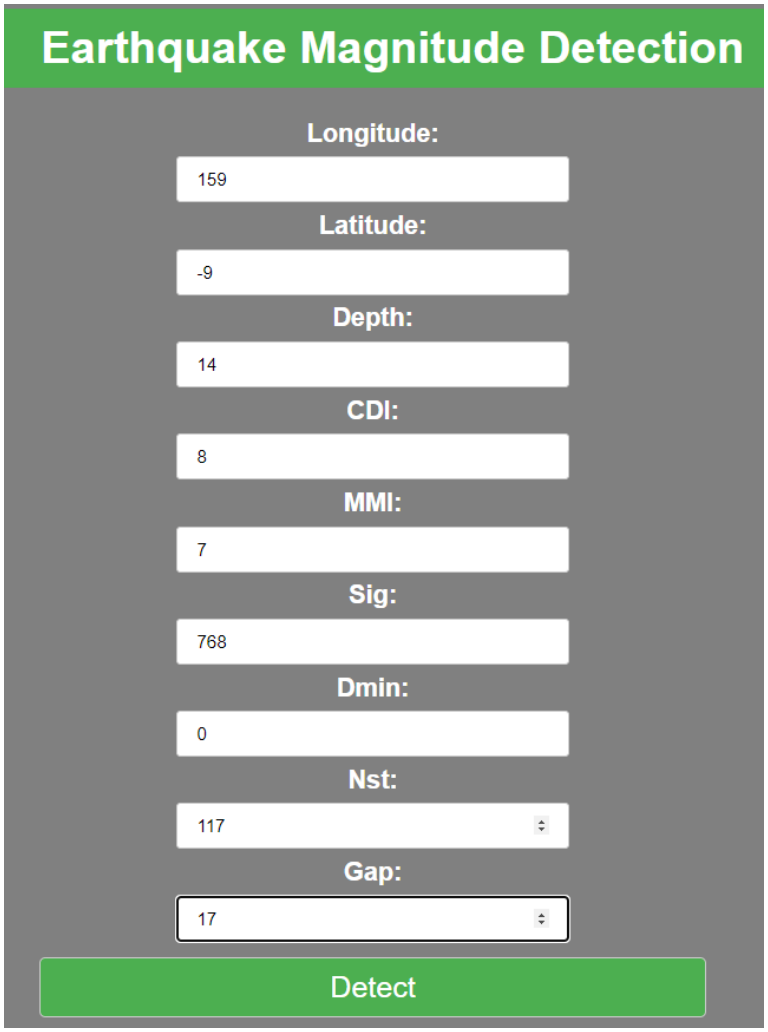


Fig 8.7: Detection of earthquake using web app UI

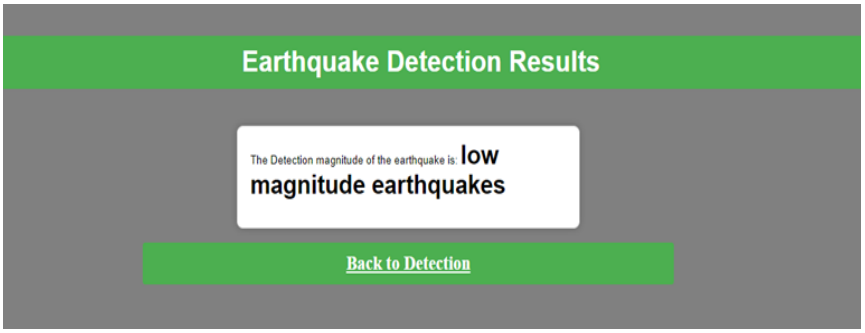


Fig 8.8: Result of the earthquake using webapp UI

CONCLUSION

Based on the evaluation of four machine learning algorithms, namely Decision Tree Classifier, Kneighbors Classifier, SVC, and Random Forest Classifier, it can be concluded that all models have performed well in detecting earthquakes. Among all the algorithms, the Random Forest Classifier algorithm achieved the highest accuracy score of 0.976923, which is the best performance compared to the other models. The Decision Tree Classifier, Kneighbors Classifier, and SVC algorithms also performed well, with accuracy scores of 0.942308, 0.923077, and 0.930769, respectively. Moreover, all the models have performed well in terms of precision, recall, and F1-score. These performance metrics are essential for earthquake detection as it requires accurate and reliable predictions to avoid false positives and false negatives. In conclusion, the Random Forest Classifier algorithm is the best model for earthquake detection based on the given dataset. However, other algorithms such as Decision Tree Classifier, K-Neighbour's Classifier, and SVC can also be used as they have also shown good performance.

REFERENCES

[1] DinkyTulsi Nandwani, 2Vanita Buradka, **"Earthquake Damange Prediction Using Machine Learning"**, © 2022 IJCRT | Volume 10, Issue 7 July 2022 | ISSN: 2320-288, www.ijcrt.org.

[2] Roxane Mallouhy, Chady Abou Jaoude ,Christophe Guyeu, Abdallah Makhoul, "**Major earthquake event prediction using various machine learning"** and author and publicationat: https://www.researchgate.net/publication/339901560.

[3] Vindhya Mudgal, 2Jayashree M Kudari, 3Ravi Chandra A, PREDICTION OF "**Earthquake Using Machine Learning Algorithms**"**,** A SURVEY PAPER ISSN :2349-5162, ESTD Year 2014, JETIR.ORG.

[4] Dr. S. Anbu Kumar1, Abhay Kumar2, Aditya Dhanraj3, Ashish Thakur, "**Earthquake Prediction using Machine Learning**" ISO 9001:2008 Certified Journal, International Research Journal of Engineering and Technology (IRJET), Issue: 05 | May 2021.

[5] Alyona Galkina, Natalia Grafeeva, , "**Machine Learning Methods for Earthquake Prediction**" CEUR–ws.org/vpl-2372/SEIM_2019_paper_31.

[6] W. Li, N. Narvekar, N. Nakshatra, N. Raut, B. Sirkeci and J. Gao. "**Seismic Data Classification Using Machine Learning**"**,** 2018 IEEE Fourth International Conference on Big Data Computing Service and Applications (Bigdata Service), Bamberg, 2018, pp. 56-63.

[7] G. Cortés, A. Morales-Esteban, X. Shang, and F. Martínez-Álvarez, "**Earthquake Prediction in California Using Regression Algorithms and Cloud-based Big Data Infrastructure,**" Computers & Geosciences, vol. 115, pp. 198-210, 2018.

[8] Pratiksha Bangar, Deeksha Gupta, Sonali Gaikwad, Bhagyashree Marekar, Jyoti Patil, "**Earthquake Prediction using Machine Learning Algorithm**" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878 (Online), Volume-8 Issue-6, March 2020

PUBLICATION

[1] MR. Nazeer Shaik, G. Akshaya, A. Manoj, B. Aravind, T. Deepthi "**THE CAVALCADE OF IMMACULATE AND DISPERATE ALGORITHMS FOR DETECTING DISTRACTED EARTHQUAKES EMPLOYING MACHINE LEARNING**" JETIR (ISSN-2349-5162) Vol. 10, Issue 4, April 2023.

JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

THE CAVALCADE OF IMMACULATE AND DISPERATE ALGORITHMS FOR DETECTING DISTRACTED EARTHQUAKES EMPLOYING MACHINE LEARNING

[1]Nazeer Shaik, [2]Akshaya.G, [3]Deepthi.T, [4]Manoj.A, [5]Aravind.B

[1]Assistant Professor, [2]UG Student, [3]UG Student ,[4]UG Student ,[5]UG Student
[2,3,4,5] Computer Science and Engineering (CSE), [1]CSE (Data Science)
[1,2,3,4,5]Srinivasa Ramanujan Institute of Technology, Anantapur, India

Abstract: An Earthquake is the sudden shaking of the surface of the earth resulting from sudden release of energy in the Lithosphere. Reliable prediction of earthquakes has numerous societal and engineering benefits. In recent years, the exponentially rising volume of seismic data has led to the development of several automatic earthquake detection algorithms through machine learning approaches. Different algorithms have been applied on earthquake detection like Decision tree, Random Forest classifier, Support vector Machine, K Nearest Neighbour Classifier.

Index Terms - Machine Learning, Earthquake, Random Forest Classifier, Decision tree, Support Vector Machine, K Nearest Neighbour Classifier.

I. INTRODUCTION

Earthquakes generally occur because of plate tectonics. While it is important to understand the nonlinear dynamics of earthquake generation process. Earthquake detection is an essential area of research that deals with identifying and analyzing seismic activity, which can cause significant destruction and loss of life. Traditional earthquake detection methods rely on seismometers, which can be expensive and not always reliable. However, with the advancements in machine learning, earthquake detection using machine learning algorithms has gained popularity. Machine learning algorithms can identify patterns in large amounts of seismic data and detect seismic activity in real-time. This has the potential to improve the speed and accuracy of earthquake detection, allowing for earlier warnings and more effective disaster response. In this context, machine learning algorithms are trained on seismic data and use various techniques to detect earthquakes.

II. LITERATURE SURVEY

Ali G. Hafez, Ahmed Abdel Azim, M. Sami Soliman & Hideki Yayama [1] have proposed a technique called P-wave picking which is one of the important steps for earthquake parameter determination. Although manual inspection of P-wave arrival timing is the most accurate method for detection but, using automated algorithm is a must to facilitate this continuous task. This algorithm generates a daily report of all events recorded by this sub-network. This algorithm can detect very small events starting from microearthquakes due to the use of multiresolution analysis (MRA) of discrete wavelet transform (DWT). Results show a high rate of successful detections of 94.6% with low false alarm rate.

Lomax, A. Michelini and D. Josipovic's [2] have suggested an investigation of rapid earthquake characterization using single station waveforms and a convolutional neural network. Effective early-warning, response and information dissemination for earthquake sand tsunamis require rapid characterization of an earthquake's location, size and other parameters. This characterization is mainly provided by real-time seismogram analysis using established, rule-based, seismological procedures. With the advent of powerful machine learning tools to make predictions from large data sets.

Omar M. Saad, Ali G. Hafez, and M. Sami Soliman [3] has proposed Deep Learning Approach for Earthquake Parameters Classification in Earthquake Early Warning System. Magnitude determination of earthquakes is a mandatory step before an earthquake early warning (EEW) system sends an alarm. Beneficiary users of EEW systems depend on how far they are located from such strong events. Therefore, determining the locations of these shakes is an important issue for the tranquillity of citizens as well. The proposed algorithm depends on a convolutional neural network (CNN) which can extract significant features from waveforms that enabled the classifier to reach a robust performance in the required earthquake parameters. The classification accuracies of the suggested approach for magnitude, origin time, depth, and location are 93.67%, 89.55%, 92.54%, and 89.50%, respectively.

H Hang Zhang 1 2, Jun Zeng 1, Chunchi Ma 1 3, Tianbin Li 1, Yelin Deng 1, Tao Song [4] has proposed a Multi-Classification of Complex Microseismical Waveforms Using Convolutional Neural Network. In this study, a micro seismic multi-classification (MMC)model is proposed based on the short time Fourier transform (STFT) technology and convolutional neural network (CNN). The real and imaginary parts of the coefficients of micro seismic data are inputted to the proposed model to generate three classes of targets. micro seismic data recorded under different geological conditions are also tested to prove the generality of the model, and a micro seismic signal with Mw ≥ 0.2 can be detected with a high accuracy. The proposed method has great potential to be extended to the study of exploration seismology and earthquakes.

Khawaja Asim, Abdul Basit, Francisco Martinez-Alvarez and Talat Iqbal [5] has detected Earthquake magnitude in Hindukush region using machine learning techniques.In the research, four machine learning techniques including pattern recognition neural network, recurrent neural network, random forest, and linear programming boost ensemble classifier are separately applied to model relationships between calculated seismic parameters and future earthquake occurrences. Here, several performance measures can be done with parameters and accuracy can be estimated.

III. PROPOSED WORK

1.Random Forest Classifier

- A random forest is a machine learning technique that's used to solve regression and classification problems. It utilizes ensemble learning, which is a technique that combines many classifiers to provide solutions to complex problems.
- A random forest algorithm consists of many decision trees. The 'forest' generated by the random forest algorithm is trained through bagging or bootstrap aggregating. Bagging is an ensemble meta-algorithm that improves the accuracy of machine learning algorithms.
- The (random forest) algorithm establishes the outcome based on the predictions of the decision trees. It predicts by taking the average or mean of the output from various trees. Increasing the number of trees increases the precision of the outcome.
- A random forest eradicates the limitations of a decision tree algorithm. It reduces the over fitting of datasets and increases precision. It generates predictions without requiring many configurations in packages (like Scikit-learn).

Features of a Random Forest Algorithm:

- It's more accurate than the decision tree algorithm.
- It provides an effective way of handling missing data.
- It can produce a reasonable prediction without hyper-parameter tuning.
- It solves the issue of over fitting in decision trees.
- In every random forest tree, a subset of features is selected randomly at the node's splitting point.

2.Decision Tree

Overall, the decision tree algorithm can help improve the accuracy of earthquake detection by identifying the most important features for classification, allowing for the creation of more complex models that capture the nuances of the seismic data. This can ultimately lead to faster and more accurate detection of earthquakes, which is critical for early warning and response systems.

The decision tree algorithm is a method used to identify important features in earthquake detection using Artificial Neural Networks (ANN). It works by creating a tree-like structure that splits the seismic data into branches based on the most informative features, helping to distinguish between earthquake and non-earthquake events. This algorithm can lead to more accurate detection of earthquakes by identifying patterns in the seismic data that may be difficult for humans to recognize. By using both ANN and decision tree algorithms, more complex models can be developed, which can improve the accuracy of earthquake detection and provide faster warnings to people in affected areas.

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules, and each leaf node represents the outcome.

3.Support Vector Machine

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

SVC works by finding the optimal boundary, or hyperplane, between two classes of data points, in this case, earthquake and non-earthquake events. The algorithm identifies the data points closest to the boundary, known as support vectors, and uses them to construct the boundary.

4.K-Nearest Neighbour

K-Nearest Neighbours (KNN) algorithm is a machine learning algorithm that can be used for various tasks, including classification and regression. In the context of earthquake detection, KNN can be used as a predictive model to identify seismic events based on their similarity to previously recorded earthquakes.

In earthquake detection, KNN can be used as a pattern recognition tool to identify seismic events that are similar to known earthquakes based on their location, magnitude, and other features. This can help in early warning systems and rapid response to earthquakes. For example, if a new seismic event occurs, KNN can be used to quickly identify if it is similar to known earthquakes in the area, and if so, issue an alert to nearby populations to take appropriate action.

IV. SYSTEM ARCHITECTURE

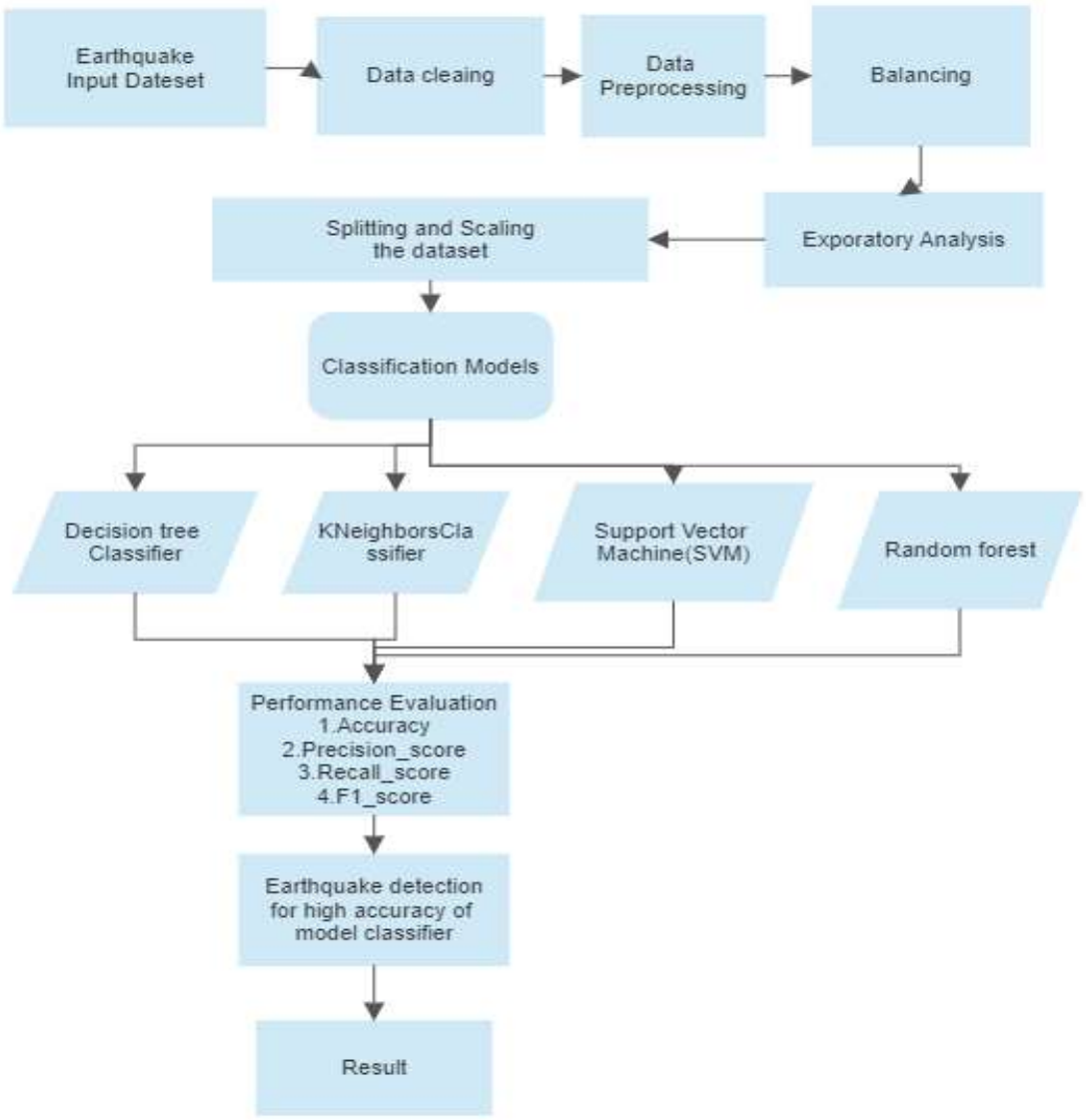


Figure.1: System Architecture

4.1. Steps Involved in Design

- Data Collection
- Data Pre-Processing
- Model Training
- Model Evaluation

4.1.1 Data Collection

Data is an important asset for developing any kind of Machine learning model. Data collection is the process of gathering and measuring information from different kinds of sources. This is an initial step that has to be performed to carry out a Machine learning project. In the present internet world these datasets are available in different websites (Ex: Kaggle, Google public datasets, Data.gov etc.) The dataset used in our project is downloaded from the Kaggle website and it contains nearly 116 records and 20 different attributes. The dataset consists of 19 independent attributes and one dependent attributes. So, the aim of the project is to predict the dependent variables using independent variables.

4.1.2 Data Preprocessing

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model. When creating a machine learning project, it is not always the case that we come across clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put it in a formatted way. So, for this, we use data preprocessing tasks. Earthquake Detection using Machine Learning Algorithms.

Preprocessing of the data consists of different kinds of steps in which analysis of the data, Data cleaning, Data encoding are part of this.

4.1.1.1 Exploratory Data Analysis

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model. When creating a machine learning project, it is not always the case that

we come across clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put it in a formatted way. So, for this, we use data preprocessing tasks. Preprocessing of the data consists of different kinds of steps in which analysis of the data, Data cleaning, Data encoding are part of this.

- Dimension reduction techniques which help create graphical display of high dimensional data containing many variables.
- Univariate visualization of each field in the raw dataset, with summary statistics.
- Bivariate visualizations and summary statistics that allows you to assess the relationship between each variable in the dataset and the target variable in the dataset and the target variable you're looking at.
- Multivariate visualizations, for mapping and understanding interactions between different fields in the data.
- This data analysis is of two types:
  a. Univariate analysis
  b. Bivariate analysis
    - Univariate analysis is the simplest form of data analysis where the data being analyzed contains only one variable. Since it's a single variable it doesn't deal with causes or relationships.
    - Bivariate data is data that involves two different variables whose values can change. Bivariate data deals with relationships between these two variables.

4.1.1.2 Filling Missing Data & Data Encoding

- The next step of data preprocessing is to handle missing data in the datasets. If our dataset contains some missing data, then it may create a huge problem for our machine learning model. Hence it is necessary to handle missing values present in the dataset.
- By calculating the mean and Mode: In this way, we will calculate the mean or Mode of that column or row which contains any missing value and will put it on the place of missing value. This strategy is useful for the features which have numeric data such as age, salary, year, etc.
- Data encoding: Since the machine learning model completely works on mathematics and numbers, but if our dataset would have a categorical variable, then it may create trouble while building the model. So, it is necessary to encode these categorical variables into numbers.
- Our dataset also consists of different categorical data in which they are encoded in this step.

4.1.3 Training the Model

In this step the model is trained using the algorithms that are suitable. Flood prediction is a kind of problem in which One variable has to be determined using some independent variables Regression model is suitable for this kind of scenario.

- A training model is a dataset that is used to train an ML algorithm. It consists of the sample output data and the corresponding sets of input data that have an influence on the output. The training model is used to run the input data through the algorithm to correlate the processed output against the sample output.
- Our project implements these algorithms like Logistic Regression, K Nearest Neighbor, Support Vector Machine, Random Forest.

4.1.4 Model Evaluation

In this step the trained model is evaluated by determining the accuracy of the model against the test data. various ways to check the performance of our machine learning or deep learning model and why to use one in place of the other. We will discuss terms like:

- Accuracy
- Recall score
- Precision score
- F1 score

Out of these we used Accuracy for evaluating our model. Accuracy is the most commonly used metric to judge a model and is actually not a clear indicator of the performance. The worst happens when classes are imbalanced balanced datasets. In such cases, other evaluation metrics such as precision, recall, and F1 score can provide a more informative picture of the model's performance. It is important to carefully choose the evaluation metrics based on the characteristics of the dataset and the problem being solved. In the case of imbalanced datasets, using accuracy alone can lead to inaccurate conclusions about the performance of a machine learning model, and it is important to consider alternative metrics such as precision, recall, and F1 score.

V. RESULTS AND DISCUSSION

5.1 Evaluation of Algorithms:

Evaluates the performance of four different machine learning algorithms, including decision tree, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Random Forest.

	Name	Accuracy	Precision_score	Recall_score	F1_score
0	DecisionTreeClassifier	0.942308	0.941710	0.942308	0.941875
1	KNeighborsClassifier	0.923077	0.934314	0.923077	0.923851
2	SVM	0.930769	0.945339	0.930769	0.932321
3	RandomForestClassifier	0.976923	0.977345	0.976923	0.976927

Table 5.1: Evaluation of Algorithms

5.2 Performance comparison of Model:

After evaluating the performance of the four machine learning algorithms, the results can be compared to determine which algorithm performs the best. For instance, the accuracy, precision, recall, and F1 score.

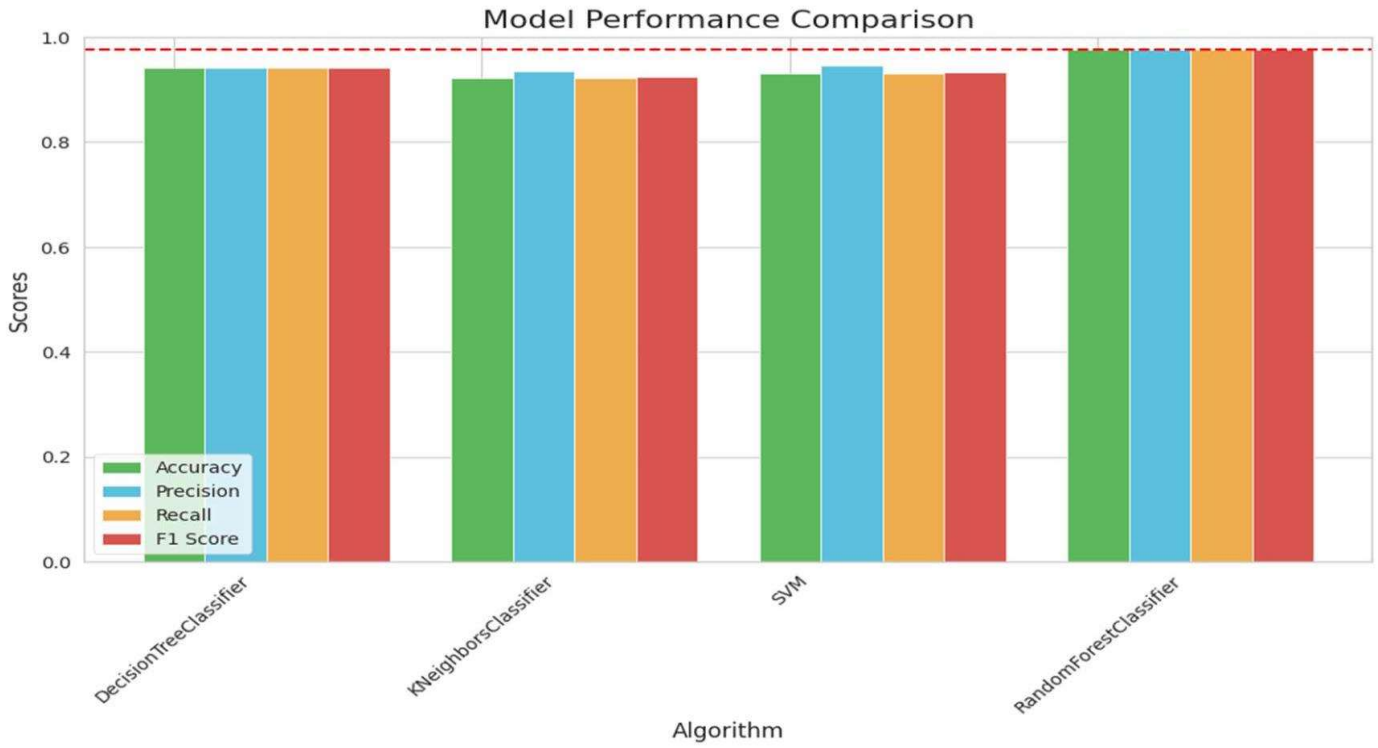


Table 5.2: Performance Comparison of Algorithms

5.3 Web app results:

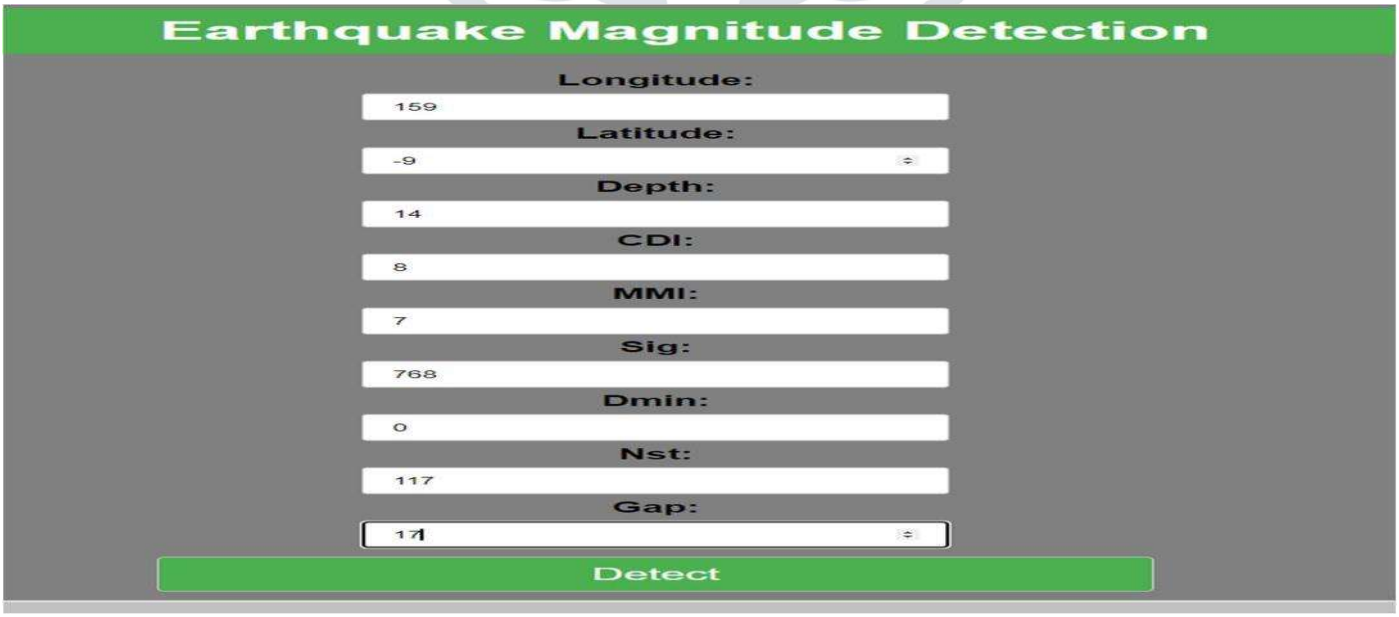


Fig 5.3: Detection of earthquake using web app UI

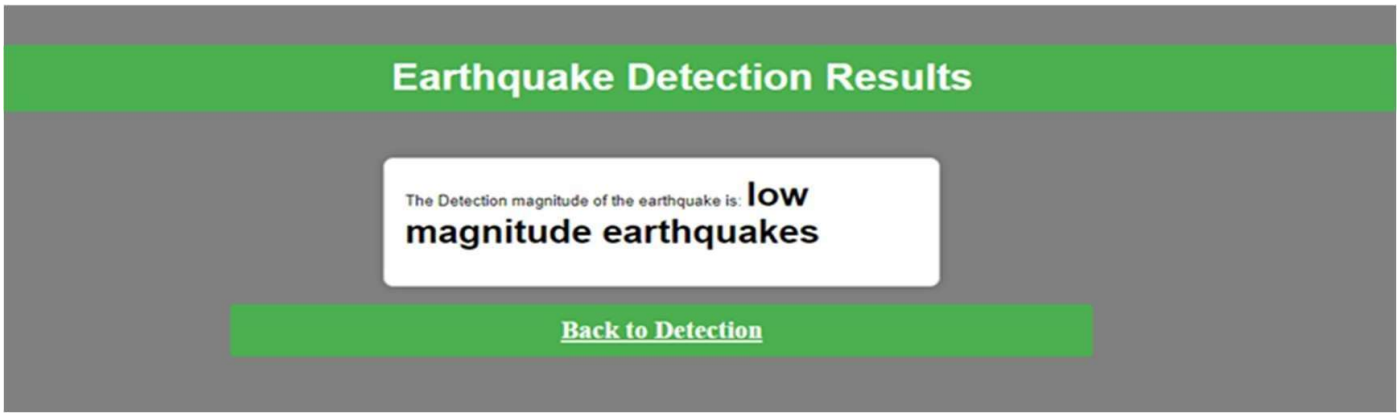


Fig 5.4: Result of the earthquake using webapp UI

V. CONCLUSION

Based on the evaluation of four machine learning algorithms, namely Decision Tree Classifier, Kneighbors Classifier, SVC, and Random Forest Classifier, it can be concluded that all models have performed well in detecting earthquakes. Among all the algorithms, the Random Forest Classifier algorithm achieved the highest accuracy score of 0.976923, which is the best performance compared to the other models. The Decision Tree Classifier, Kneighbors Classifier, and SVC algorithms also performed well, with accuracy scores of 0.942308, 0.923077, and 0.930769, respectively. Moreover, all the models have performed well in terms of precision, recall, and F1-score. These performance metrics are essential for earthquake detection as it requires accurate and reliable predictions to avoid false positives and false negatives. In conclusion, the Random Forest Classifier algorithm is the best model for earthquake detection based on the given dataset. However, other algorithms such as Decision Tree Classifier, K-Neighbour's Classifier, and SVC can also be used as they have also shown good performance.

VI. REFERENCES

1. Dinkytulsi Nandwani, 2vanita Buradka, Earthquake Damage Prediction Using Machine Learning, © 2022 Ijcrt | Volume 10, Issue 7 July 2022 | Issn: 2320-288, Www.Ijcrt.Org.

2. Roxane Mallouhy, Chady Abou Jaoude ,Christophe Guyeu, Abdallah Makhoul, Major earthquake event prediction using various machine learning and author profiles for this publication at: https://www.researchgate.net/publication/339901560.

3. Vindhya Mudgal, 2jayashree M Kudari, 3ravi Chandra A,Prediction Of Earthquakes Using Machine Learning Algorithms: A Survey Paper" Issn :2349-5162 ,Estd Year 2014,Jetir.Org.

4. Dr. S. Anbu Kumar1, Abhay Kumar2, Aditya Dhanraj3, Ashish Thakur, Earthquake Prediction using Machine Learning, ISO 9001:2008 Certified Journal, International Research Journal of Engineering and Technology (IRJET), Issue: 05 | May 2021.

5. Alyona Galkina, Natalia Grafeeva, Machine Learning Methods for Earthquake Prediction: a Survey,CEUR –ws.org/vpl-2372/SEIM_2019_paper_31.

6. W. Li, N. Narvekar, N. Nakshatra, N. Raut, B. Sirkeci and J. Gao. "Seismic Data Classification Using Machine Learning", 2018 IEEE Fourth International Conference on Big Data Computing Service and Applications (BigDataService), Bamberg, 2018, pp. 56-63.

7. G. Cortés, A. Morales-Esteban, X. Shang, and F. Martínez-Álvarez, "Earthquake Prediction in California Using Regression Algorithms and Cloud-based Big Data Infrastructure," Computers & Geosciences, vol. 115, pp. 198-210, 2018.

8. Pratiksha Bangar, Deeksha Gupta, Sonali Gaikwad, Bhagyashree Marekar, Jyoti Patil, "Earthquake Prediction using Machine Learning Algorithm" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878 (Online), Volume-8 Issue-6,March 2020.

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Nazeer Shaik

In recognition of the publication of the paper entitled

The Cavalcade of Immaculate and Disparate Algorithms for Detecting Distracted Earthquakes Employing Machine Learning.

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 10 Issue 4 , April-2023 | Date of Publication: 2023-04-24

EDITOR EDITOR IN CHIEF

JETIR2304835 Research Paper Weblink http://www.jetir.org/view?paper=JETIR2304835 Registration ID : 511963

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Akshaya.G

In recognition of the publication of the paper entitled

The Cavalcade of Immaculate and Disparate Algorithms for Detecting Distracted Earthquakes Employing Machine Learning.

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 10 Issue 4 , April-2023 | Date of Publication: 2023-04-24

EDITOR EDITOR IN CHIEF

JETIR2304835 Research Paper Weblink http://www.jetir.org/view?paper=JETIR2304835 Registration ID : 511963

An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor Calculate by Google Scholar and Semantic Scholar | AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal Indexing in All Major Database & Metadata, Citation Generator

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Manoj.A

In recognition of the publication of the paper entitled

The Cavalcade of Immaculate and Disparate Algorithms for Detecting Distracted Earthquakes Employing Machine Learning.

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 10 Issue 4 , April-2023 | Date of Publication: 2023-04-24

EDITOR EDITOR IN CHIEF

JETIR2304835 Research Paper Weblink http://www.jetir.org/view?paper=JETIR2304835 Registration ID : 511963

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Aravind.B

In recognition of the publication of the paper entitled

The Cavalcade of Immaculate and Disparate Algorithms for Detecting Distracted Earthquakes Employing Machine Learning.

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 10 Issue 4 , April-2023 | Date of Publication: 2023-04-24

EDITOR EDITOR IN CHIEF

JETIR2304835 Research Paper Weblink http://www.jetir.org/view?paper=JETIR2304835 Registration ID : 511963

An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact Factor Calculate by Google Scholar and Semantic Scholar | AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal Indexing in All Major Database & Metadata, Citation Generator

Journal of Emerging Technologies and Innovative Research
An International Open Access Journal Peer-reviewed, Refereed Journal
www.jetir.org | editor@jetir.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of

Journal of Emerging Technologies and Innovative Research (ISSN : 2349-5162)

Is hereby awarding this certificate to

Deepthi.T

In recognition of the publication of the paper entitled

The Cavalcade of Immaculate and Disparate Algorithms for Detecting Distracted Earthquakes Employing Machine Learning.

Published In JETIR (www.jetir.org) ISSN UGC Approved (Journal No: 63975) & 7.95 Impact Factor

Published in Volume 10 Issue 4 , April-2023 | Date of Publication: 2023-04-24

Parisa P
EDITOR

EDITOR IN CHIEF

JETIR2304835 Research Paper Weblink http://www.jetir.org/view?paper=JETIR2304835 Registration ID : 511963