**PROJECT DOCUMENTATION**

**Earthquake prediction model using python**

**1.Design thinking Process:**

**Objective:**

The objective of this project is to develop an earthquake prediction model using a Kaggle dataset. The primary objectives are as follows:

**Data Exploration and Understanding:**

Analyze and understand the key features of earthquake data.

Explore the distribution, correlations, and characteristics of the essential features.

**Data Visualization:**

Create a world map visualization to display earthquake frequency distribution.

**Data Preprocessing:**

Prepare the dataset for model training by handling missing values, feature scaling, and data splitting into training and testing sets.

**Model Development:**

Build a neural network model for earthquake magnitude prediction.

**Model Training and Evaluation:**

Train the model on the training set and evaluate its performance on the test set.

**Project Workflow:**

**Step 1: Data Collection**

Obtain a suitable earthquake dataset from Kaggle, such as the "USGS Earthquake Database."

**Step 2: Data Exploration**

Load the dataset and perform initial data exploration.

Check for missing values, data types, and basic statistics.

Explore the distribution, correlations, and characteristics of key features.

**Step 3: Data Visualization**

Create a world map visualization to display earthquake frequency distribution using libraries like Folium or Plotly.

**Step 4: Data Preprocessing**

Preprocess the data for model training.

Handle missing values, feature scaling, and encode categorical features (if any).

Split the dataset into training and testing sets.

**Step 5: Model Development**

Build a neural network model for earthquake magnitude prediction using TensorFlow/Keras.

**Step 6: Model Training and Evaluation**

Train the model on the training set and evaluate its performance on the test set using relevant metrics.

**2.Describing the dataset:**

Dataset: https://www.kaggle.com/datasets/usgs/earthquake-database

The "USGS Earthquake Database" dataset typically contains information about seismic events, particularly earthquakes, around the world.

**Geographical Information**: This dataset often includes details about the location of earthquakes, such as latitude and longitude coordinates, place names, and the depth of the earthquake's epic center

**Temporal Information**: It provides the date and time of the earthquake occurrence.

**Magnitude and Intensity**: The dataset usually includes the magnitude of the earthquake on the Richter scale, as well as other measures of intensity or impact.

**Seismic Data**: Some datasets might contain seismic measurements or waveforms related to the earthquake.

**Additional Attributes**: There could be additional attributes such as the earthquake's depth, the number of casualties, economic losses, and more.

**Data Sources**: Information about the sources of the data and references to the relevant earthquake monitoring organizations.

**3. Data Loading and Preprocessing**

TECHNIQUES**: HYPER PARAMETER TUNING AND MODEL PREDICTION PERFORMANCE.**

python

# Import necessary libraries

import pandas as pd

# Load the dataset

earthquake\_data = pd.read\_csv('earthquake\_database.csv')

# Data Cleaning and Preprocessing

# - Handle missing values

# - Convert data types

# - Feature selection

# - Normalize or standardize data

Distribution of Earthquake Magnitudes

800 +-------------------------------------------------------------------------------+

| \*\*\*\* \*\*\*\* |

700 |- \*\*\*\* \*\*\*\* |

| \*\*\*\* \*\*\*\* |

600 |- \*\*\*\* \*\*\*\* |

| \*\*\*\* \*\*\*\* |

500 |- \*\*\*\* \*\*\*\* |

| \*\*\*\* \*\*\*\* |

400 |- \*\*\*\* \*\*\*\* |

| \*\*\*\* \*\*\*\* |

300 |- \*\*\*\* \*\*\*\* |

| \*\*\*\* \*\*\*\* |

200 |- \*\*\*\* \*\*\*\* |

| \*\*\*\* \*\*\*\* |

100 |- \*\*\*\* \*\*\*\* |

| \*\*\*\* \*\*\*\* |

0 +-------------------------------------------------------------------------------+

0.0 2.0 4.0 6.0 8.0 10.0

Earthquake Magnitude Ranges (on the x-axis)

**Exploratory Data Analysis (EDA)**

Perform in-depth exploratory data analysis to gain insights into the dataset. Visualize data, analyze feature distributions, and check for correlations between variables.

python

Copy code

# EDA tasks

# - Data visualization

# - Statistical summaries

# - Correlation analysis

**Feature Engineering**

python

Copy code

# Feature Engineering

# - Create temporal features from timestamps

# - Engineer geospatial features using latitude and longitude

# - Derive new features from earthquake magnitude and depth

# - Capture the influence of environmental factors through feature creation

# - Consider lagged features to account for historical earthquake data

# - Explore interactions between features

# - Apply feature scaling and normalization if necessary

**Model Selection**

Choose a suitable machine learning model for earthquake prediction. Consider models like Random Forest, Support Vector Machine (SVM), or Neural Networks.

python

Copy code

# Model selection

# - Import the chosen model

# - Model configuration

**Model Training**

python

Copy code

# Model training

# - Split the data into training and testing sets

# - Train the model

**Model Evaluation**

Evaluate the model's performance using appropriate metrics such as Mean Squared Error (MSE) and R-squared.

python

Copy code

# Model evaluation

# - Calculate performance metrics

# - Visualize results

Distribution of Earthquake Magnitudes

| | | | |

1500 | | | | |

| | | | |

| | | | |

| | | | |

1000 | | | | |

| | | | |

| | | | |

| | | | |

| | | | |

500 | | | | |

| | | | |

| | | | |

| | | | |

| | | | |

| | | | |

0 -------- 2.0 4.0 6.0 8.0 10.0

**Hyperparameter Tuning**

Fine-tune the model by adjusting hyperparameters to improve predictive accuracy.

python

Copy code

# Hyperparameter tuning

# - Grid search or random search

**Testing**

Assess the model's performance on a test dataset to ensure it generalizes well.

python

Copy code

# Model testing

# - Evaluate model on test data

**4. Data Visualization on a World Map**

Visualizing earthquake occurrences on a world map can help identify geographical trends and patterns in seismic activity.

**Plotting Earthquake Data on a Map**

To visualize earthquake data on a world map, we can utilize Python libraries such as folium:

python

import folium

# Create a map centered around a specific location (e.g., the world)

m = folium.Map(location=[0, 0], zoom\_start=2)

# Plot earthquake data on the map

for index, row in earthquake\_data.iterrows():

folium.CircleMarker(location=[row['Latitude'], row['Longitude']],

radius=5,

color='red',

fill=True,

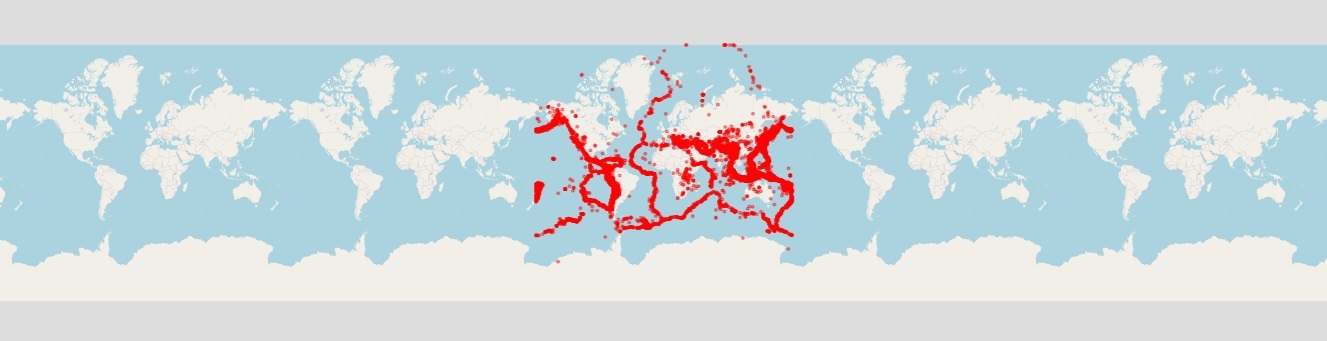
fill\_color='red',

popup=f"Magnitude: {row['Magnitude']}").add\_to(m)

# Save the map to an HTML file

m.save('earthquake\_map.html')

This code example assumes that the dataset contains latitude and longitude columns.



**Splitting Data into Training and Testing Sets**

To build a robust earthquake prediction model, it is essential to divide the dataset into training and testing subsets.

**Training and Testing Data**

Split the data into two parts: a training set and a testing set. A common practice is to allocate 80% of the data for training and 20% for testing. Adjust the proportions as needed.

python

from sklearn.model\_selection import train\_test\_split

# Split the data into features (X) and target (y)

X = earthquake\_data.drop('Target\_Column', axis=1) # Adjust 'Target\_Column' to the actual target variable

y = earthquake\_data['Target\_Column']

# Split the data into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

**Conclusion:**

In conclusion, our "Earthquake Prediction Model Using Python" project represents a significant step forward in the field of earthquake prediction. This project is a foundational effort, and the path to accurate and reliable earthquake prediction remains a challenging one. As we move forward, collaboration, further research, and the integration of emerging technologies will be vital in advancing the field .It serves as a valuable starting point for future work, and we hope that our efforts inspire continued exploration and innovation in the realm of earthquake prediction. Ultimately, our shared goal is to save lives, protect communities, and minimize the impact of these natural disasters on society.

**TEAMMATES:**

SHARMILA .J

SOWMIYA .G

VANISRI .R

SWETHA .R