Predictive Modeling and Simulation Techniques for Landslide Risk Management

by

Chowdhury Mohammad Mutamir Samit
20201223
Arian Wazed
20301039

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
June 2025

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Chowdhury Mohammad Mutamir Samit
20201223

Arian Wazed
20301039

Approval

The thesis/project titled "Predictive Modeling and Simulation Techniques for Landslide Risk Management" submitted by

1. Chowdhury Mohammad Mutamir Samit(20201223)
2. Arian Wazed(20301039)

Of Summer, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on June, 2025.

Examining Committee:

Supervisor:
(Member)

MR. Annajiat Alim Rasel
Senior Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
UG Thesis Coordinator
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Landslides are common natural disasters in the hilly areas, inflicting significant damage to both the human lives and economy. Unlike other severe disasters like floods, earthquakes etc. landslides do noticeably have a significant impact on the development initiatives. Landslides are regulated by different triggering events, which makes it impossible to forecast their exact mechanism. During the last decade, researchers have focused on using machine learning to forecast landslides. The aim of our project is to estimate the probability of landslides. In our project we will use a set of 8 features to train the model and forecast landslides. The acquired data was studied by data count ,correlation matrix and distribution of feature data.We will be analyzing the biggest landslides and find the main reasons responsible behind these landslides. The dataset will be used to test various machine learning algorithms and examine a variety of factors and visualizations.We will then examine the models to determine which performs well and have better prediction accuracy.

Keywords: Landslides, Machine learning, correlation matrix, data count, machine learning algorithms, prediction accuracy.

Acknowledgement

We are grateful to Almighty Allah for allowing us to our project without any significant setbacks. Second, thanks to MR. Annajiat Alim Rasel Sir, our supervisor, for his thoughtful counsel and support throughout our effort. He was always there to support us. It also may not have been possible without the support of our parents during our journey. With your kind prayers and support, we are almost done with our degree.

Table of Contents

List of Figures

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

AUC Area Under The Curve

BMI Body Mass Index

CNN Convolutional Neural Network

KNN K-Nearest Neighbors

$LSTM$ Long Short-Term Memory Networks

MAE Mean Absolute Error

ML Machine Learning

$MTCNN$ Multi-Task Convolutional Neural Network

RF Random Forest

$RMSE$ Root Mean Square Error

Chapter 1

Introduction

Landslides are the movement of hillslide materials such as soil, debris and rock due the gravity and pressure created by water .[7] Landslides are common natural disaster in the hilly areas, which inflicts a significant damage to human lives and the economy. [13].
Landslides are common calamities in many Asian countries. Unlike the earthquakes, floods, landslides often have a minimal effect on the development of projects and only damages the small areas. As a result of that it has got less attention among Asian countries, including Bangladesh.
Predicting the location and occurrence timing of landslides in specific areas are significantly difficult in natural hazards study and alleviation. [7] In the past, attempts were made to forecast landslides in order to handle the disasters more effectively. Landslides are regulated by various triggering events , making it impossible to forecast their exact mechanism. [13].
Our goal is to anticipate the likelihood of landslides in Rangamati Hill Tracts and provide the early warnings to prevent loss of lives and valuable assets. Real-time monitoring is the primary focus for this project.

1.0.1 Research Problem

Landslides are a frequent natural disaster in Rangamati Hill Tracts, posing a substantial risk to human lives.Despite being a common occurrence it has received little attention. As the population grows and our societies become more diverse, the economic and social costs of such events are more likely to rise up unless adequate consideration is given at an early stage of it, as increased anthropogenic activity in hilly areas will exacerbate the communities already established vulnerability. The aim of this project is to develop a machine learning model that, by merging past real-time data, can anticipate the chances of landslides in real time. Therefore, the question this study is trying to answer is:
"Which machine learning algorithm predicts the land slides with best accuracy and can model successfully enable real-time monitoring and warning systems?"
This study will answer the following questions through training different machine learning models.

1.1 Research Objective

The idea is to develop an intelligent solution that will anticipate the likelihood of landslides in Rangamati Hill Tracts and provide the early warnings to prevent loss of lives and valuable assets. Real-time monitoring is the primary focus for this project.

1. **Data Collection and Integration:** Gather historical landslide data and real-time information from satellite imagery and weather APIs to predict landslide probability in specific locations.
2. **Real-time Visualization of Landslide Risk:** Provide real-time visualization of high-risk areas using interactive maps and display key factors contributing to landslide risk, such as rainfall and ground conditions.
3. **Deployment of a User-Friendly Web Application:** A user-friendly web application will be deployed to monitor landslide risk, offering real-time alerts, visualization, and customizable monitoring features for users.

Chapter 2

Literature Review

2.0.1 Introduction

Previously we have provided a brief overview of the work, highlighting its significance and contributions. Numerous researchers have been employing machine learning and artificial intelligence to predict landslides for some time. In this section, we present the framework we planned and review important related papers. We also review the main conclusions drawn from these research and the techniques that were employed to get at those conclusions. To obtain a better understanding of the present status of research in this field, the results of these scholarly works are analyzed.

2.0.2 Related Researches

In order to predict shallow landslides in Bijar City, Iran, Shizadi et al. [13]. developed an ensemble model called Alternating Decision Tree (ADTree) that makes use of Bagging Multiboost, Random Subspace and Rotation Forest. Various sample sizes and raster resolutions can be employed with this model
In Lishui City, China, Wang et al.[15]. utilized the Synthetic Minority Oversampling Technique (SMOTE) to achieve sample balance for landslides. To create excellent landslide susceptibility maps, they employed machine learning techniques (ANN, LR, SVM and RF).
F. Ren and X. Wu [8]. used the RBF- SVN model to forecast displacement based on reservoir, groundwater level and precipitation in their study of the Baishuihe landslide in the Three Gorges region. With little inaccuracy, their refined model was able to predict outcomes with great accuracy.
J. Tingyao and W. Dinglong [6]. used the K2 method to create a Bayesian Network model that estimates landslide stability while taking cohesion and slope angle into consideration.
Xiao, Zhang and Peng [14]. predicted landslide susceptibility based on variables like elevation, slope and precipitation using data-driven algorithms (SVM, DT, LSTM, BPNN). AT 81.2%, LSTM exhibited the highest prediction accuracy.
Agarwal et al. [10]. employed synthetic oversampling approaches to address class imbalance in landslide datasets. In terms of landslide prediction, they discovered that Random Forest outperformed the other models (NN, LR, SVM and DT).

2.0.3 Satellite Image

Images of the Earth or other planets taken by artificial satellites are referred to as satellite imagery. These images offer valuable insight into the global conditions, particularly over oceans where data is often lacking. Essentially, they act as the "eye in the sky", revealing what cannot be directly observed or measured. Despite their significance, satellite images are often taken for granted. However, they provide critical "first-hand" data that can be analyzed with minimal room for error. [19].

2.0.4 MODIS/006/MCD12Q1 Dataset

The MCD12Q1 V6 product provides global ground cover types derived from six different classification schemes at yearly intervals (2001-2016). It is estimated by supervised classification of MODIS Terra and Aqua reflectance data. By further post-processing including prior knowledge and ancillary information, the supervised classifications are improved even more [20]

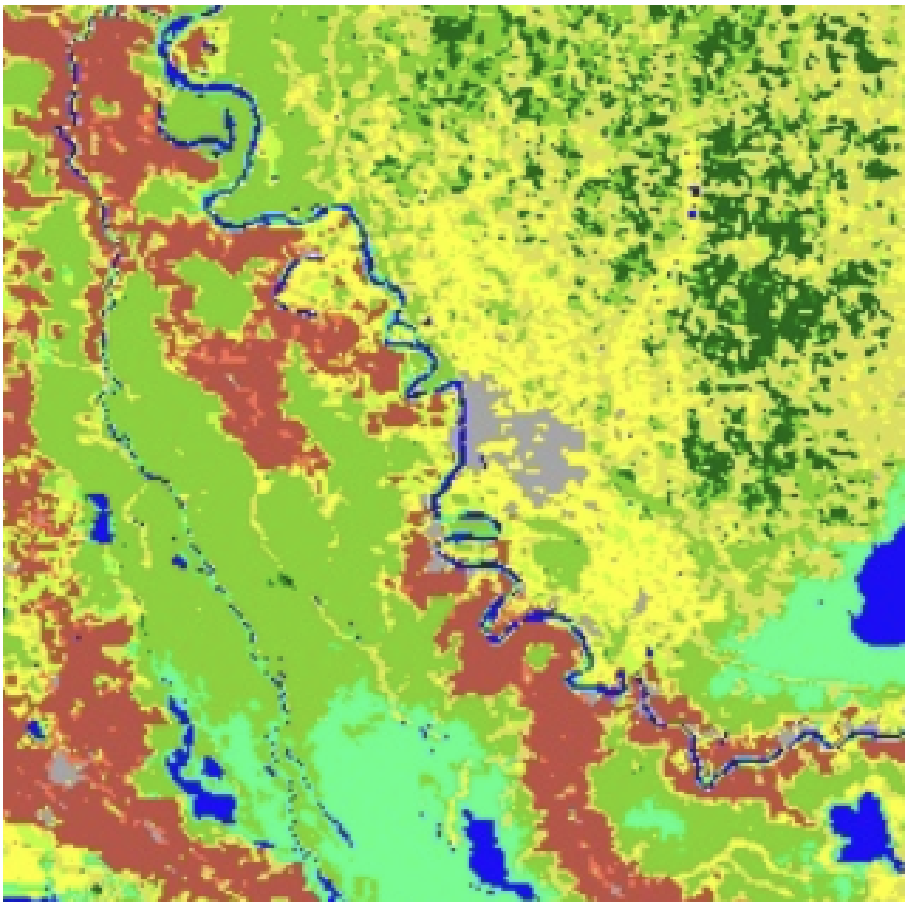


Figure 2.1: MODIS/006/MCD12Q1 Image

2.0.5 USGS/SRTMGL1_003 Dataset

The USGS/SRTMGL1_003 dataset provides elevation data at a 30-meter resolution. This dataset is crucial for analyzing terrain features such as elevation, slope, and aspect, which are essential for landslide prediction. [22].

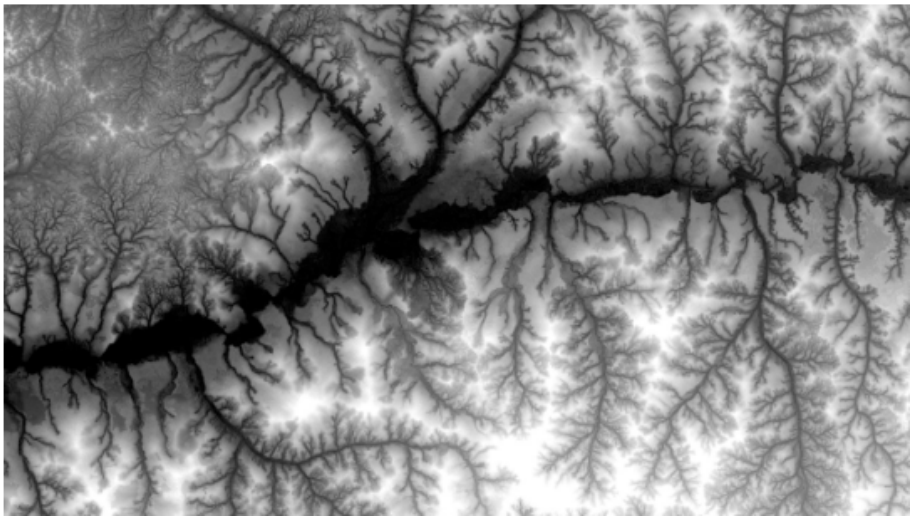


Figure 2.2: USGS/SRTMGL1_003 Image

2.0.6 LANDSAT/LC08/C01/T1_TOA Dataset

Collection Landsat 8, 1 calibrated top-of-atmosphere (TOA) reflectance from Tier 1. Extracting the picture metadata to use in calibration coefficients [23].

Figure 2.3: LANDSAT/LC08/C01/T1_TOA Image

2.0.7 MERIT/Hydro/v1_0_1 Dataset

MERIT Hydro: A new high resolution global map of flow direction and water bodies. MERIT DEM elevation data and water body datasets version 1.0.3 were used to create MERIT Hydro, a new high resolution global flow direction map at 3 arc seconds (approximately 90 m at the equator) (G1WBM, GSWO, OpenStreetMap). The result of a new algorithm to extract real inland basins and to separate river networks from noise originating from errors in the elevation data was MERIT Hydro. After only minor manual editing the resulting hydrography map is in general in good agreement with existing quality controlled river network datasets regarding the flow accumulation area and river basin configuration. The comparison of river streamlines was done realistically with current satellite derived river channel data for the world. The reliability of the found global river networks is also supported by the fact that 90 percent of the GRDC gauges have a relative error in the drainage region of less than 0.05. Most of the uncertainties in the flow accumulation region were observed in the dry river basins with depressions which at times were linked to form wrong watershed boundaries.
MERIT Hydro has better geographical coverage (between 90N and 60S) and representation of small streams compared to existing global hydrography datasets because of better availability of high-quality baseline geospatial data [21].

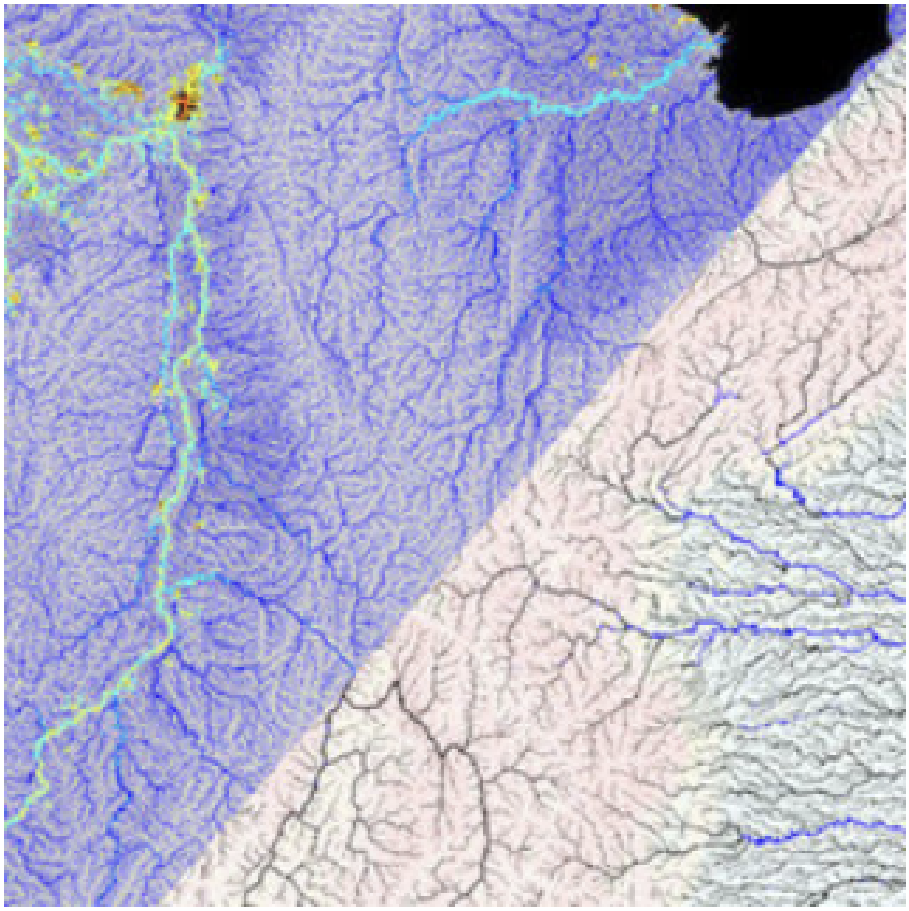


Figure 2.4: MERIT/Hydro/v1_0_1 Image

2.0.8 JAXA Climate Rainfall Watch

The importance of tracking extreme precipitation from space is widely recognized, particularly in areas where it is difficult or impossible to take land-based measurements. In response to this challenge, JAXA developed the Global satellite Mapping of Precipitation (GSMaP) under the Global Precipitation Measurement (GPM) program. The GSMaP Near real-time Rainfall Product is provided by JAXA under the World Meteorological Organization's (WMO) Space-based Weather and Climate Extremes Monitoring (SWCEM) initiative . Drawing on its experience with SWCEM, JAXA's Earth Observation Research Center (EORC) launched the "JAXA Climate Rainfall Watch" website, which uses GSMaP data to monitor and provide updates on severe heavy rainfall and drought globally. With that, users may easily track worldwide extreme weather events and environment by observing cumulative rainfall on a temporal scale as well as indices connected to extreme heavy rainfall and drought. [25].

2.0.9 Google Earth Engine API

Google Earth Engine API is a cloud based application through which an extensive library of satellite imagery and geospatial data can be accessed. It enables users to perform geospatial analysis, create custom resources and share their work. The API allows processing of large scale remote sensing and vector data to address issues like drought, natural disasters, food security, water management and climate monitoring. Its cloud architecture supports scalable computing for research and analysis.[27][24].

2.0.10 Google Map API

The Google Map Platform provides a range of SDKs and APIs that let developers easily include Google Maps into websites and mobile applications. With the help of the amazing Google Maps API, users may embed Google Maps straight into their websites. It offers choices for customizing the map's appearance to fit the theme of the website and add pertinent details that users may find useful.

2.0.11 Machine Learning Algorithms

K-Nearest Neighbors (KNN)

The K-Nearest Neighbors are one of the simplest yet most fundamental classification algorithms in machine learning. It has seen broad applications in several supervised learning tasks regarding data processing, pattern recognition, and intrusion detection. The KNN algorithm, being non-parametric in nature, assumes nothing regarding the distribution of the underlying data. It assigns a given data point, according to the algorithm, into one of several pre-defined categories based on the proximity to the other previously categorized points. Commonly, measurements of distance between two points are made using the Euclidean distance formula.

Decision Trees (DT)

Decision trees are supervised algorithms used mainly for classification and regression applications. Using this approach, the data is partitioned into sunsets based on

feature values, and a tree structure is developed to represent the model. Class labels come from the leaf nodes, whereas internal nodes denote features. Probably, decision trees are among the most valuable and popular tools of machine learning due to their simplicity and interpretability.

Support Vector Machine (SVM)

The Support Vector Machines [2], in general, are supervised algorithms that can be used for both classification and regression problems. In the case of plotting data points, SVM depends on an n-dimensional space, where n defines the number of features. Then, the algorithm creates a hyper-plane or n-1 dimensions to separate the data into classes. A "kernel trick" transformation is used by the SVMs to transform the data into an optimal boundary for classification. Approach that works quite well for complex problems of classification.

Linear Regression (LR)

Linear regression [4] uses a straight line to best fit actual data in predictive modeling of the relationship between two variables. One variable is an independent variable and the other is a dependent variable. The model helps determine whether a significant association exists between the independent and dependent variables but provides no implication of causality. Scatterplots and correlations coefficients are generally made to visualize and describe the nature of the relationship between the two variables.

Random Forest

The Random Forest is supervised learning and one of the strongest machine learning algorithms. It can be applied in the field of machine learning for the regression and classification tasks. This technique is an ensemble learning basis. Ensemble learning is a kind of method for combining different classifiers in order to solve a problem that is hard and make the model's accuracy high. The term "forest" comes from the fact that the algorithm generates a set of decision trees for a prediction task. It uses a projection of all the decision trees rather than just one and makes a prediction about the outcome based on a plurality of votes. The more trees in the landscape, the more accurate it becomes, avoiding overfitting. [3].

Bayesian Regression (BR)

Bayesian Regression [12] models the relationships between variables using probability distribution rather than point estimates. Rather than computing the optimal value of model parameters, the Bayesian methods examine a posterior distribution. This probabilistic perspective on model parameters is richer compared with traditional regression models.

Neural Network (NN)

Neural Networks truly imitate the composition and organization of the human brain, aiming at the patterns within the data. Neural networks comprise interconnected layers of neurons with the ability to adapt to various inputs of data. Neural networks

are a major part of the applications concerning artificial intelligence, working well on complex tasks such as image recognition, speech processing, and trading systems.[16]

AdaBoost

AdaBoost, or Adaptive Boosting, is a machine learning boosting ensemble method. It recalculates the weights of each instance; the larger the weight is, the more chances an instance, which was wrongly tagged, will have to be correct. Hence, it's called adaptive boosting. Boosting tries to reduce the variance and bias in a supervised learning environment. It's based on an intuitive, sequential growth of learners. All other learners, except the first one, are derived from previous learners. In other words, bad students are transformed into good ones. While the Adaboost approach operates based on the same principle as boosting, it does it a bit differently. [26].

Gradient Tree Boosting

The boosting method is the origin of the machine learning technique called gradient boosting. The technique is based on the intuition that the average prediction error is minimized by combining the best next model with the previous ones. The concept, to minimize error, is deriving the target results for the next model. It calculates the target value for each data case according to how much changing its prediction would reduce the total prediction error:

- If a slight increase in one case's estimation leads to a large decrease in error, that case's next target value is very high. The error will decrease if the current model's predictions are towards its targets.
- If a small increment in the prediction of a case makes no difference in error, the next target output for the case is zero. Changing this prediction makes little difference in the error.

Gradient boosting gets its name from the fact that the target values for each case are defined as the gradient of the error with respect to the prediction. [17]

2.0.12 Soft Voting Ensemble Model (Hybrid Model)

Soft Voting Ensemble Hybrid Model is a sophisticated hybrid machine learning method that blends the power of several different individual classifiers to enhance prediction performance. In this model, heterogeneous base learners like Logistic Regression, Random Forest, and Gradient Boosting are trained on identical data. The hybrid model does not use the prediction of one model but averages the predicted probabilities of all the base learners and utilizes it. This approach, termed soft voting, uses the highest average probability of all the models to make the final prediction, yielding more stable and accurate results. Heterogeneity of models enables the hybrid system to compensate for the deficiency in specific algorithms, making it particularly suitable for complicated classification problems. This strategy of combination is especially effective when the base models are calibrated and probabilistic and therefore enhance robustness and prevent overfitting.

2.0.13 Web Tools

2.0.14 Frontend Tool

HTML

The most common markup language for texts to be displayed in a web browser is HTML, HyperText Markup Language. Web browsers translate the HTML documents into multimedia web pages upon retrieval from a web server or locally stored files. HTML (HyperText Markup Language) is the most used markup language in documents destined to be viewed with a web browser. Web browsers online take an HTML document fetched from a web server or that one has stored on the device, and change it into an animated web page. HTML was invented for hints about presenting documents and semantic structuring of the web page. HTML building blocks are its elements, which are structures in which a photographer can upload a photo along with other stuff inside the opened tab, an interactive type among many others. HTML provides a facility for creating structured documents by marking up structural semantics of text like headings, columns, links, quotes, and many other artifacts. Browsers do not visualize the HTML tags; they are simply used to read the text on the website.[29]

CSS

The way text written in a markup language, like HTML is presented, is described by the style sheet language CSS. It is one of the three components that make up the World Wide Web, together with HTML and JavaScript. It's a style sheet that separates the presentation from the text and allows changing of the interface, colors, and fonts. This prevents redundancy and duplication of structural content, which means the CSS file can be cached so that loading between pages with similar formats and files is much quicker. Separation helps in improving the usability of the content, and it also allows the specifier to have more consistency and control over the presentation characteristics specified in it. In order to improve content accessibility, give more freedom and control in defining presentation features and enable multiple web pages, it encourages several web pages to share formatting by having appropriate CSS in a distinct CSS file..[28]

2.0.15 Backend Tool

Flask

Flash is a microweb framework written in python. It is referred to as a microframework since it doesn't require the use of any particular tool or repository. A database abstraction layer, type validation and other components that rely on third-party libraries to carry out some simple tasks are not included. Extensions can be used to implement application features inside Flask as if it were built-in. Object-relational mappers, form validation, upload management, transparent authentication methods and other framework-related extensions are among the available extensions. The following constitutes Flask:

1. **WSGI:** The industry standard for building Python web applications is this.

The Network Service Gateway Interface is a standard for creating a universal interface between a web server and a web application.

2. **Werkzeug:** This is a WSGI toolkit and it manages the request, the artifact responses among other core operations. Therefore it is your base upon which you can put your web application. Werkzeug forms the very back bone of the Flask system.

3. **Jinja2:** Jinja2 is one famous template engine for Python. A web template framework that integrates with a data source of some sort is used to generate dynamic web pages.

Chapter 3

Framework

3.1 Work Plan

Data is a vital component of a machine learning system since it feeds the model. As traditional machine learning models are sensitive to feature selection and data preprocessing techniques, feature selection is a crucial step before model training. The historical data is first gathered preprocessed and then separated into sets for testing and training. After analyzing the data, visuals made it easier for us to comprehend how different features affect the occurrence of landslides. Several machine learning models are trained using the preprocessed data that had been divided for training. Based on how well it performed in comparison to other models, a final model was chosen, and it is then used to make predictions using current data that Google Earth Engine had collected from satellite imagery. Both frontend and backend technologies are used in the development of a web application. To deliver the most accurate prediction results, the chosen model will be included into the online application.

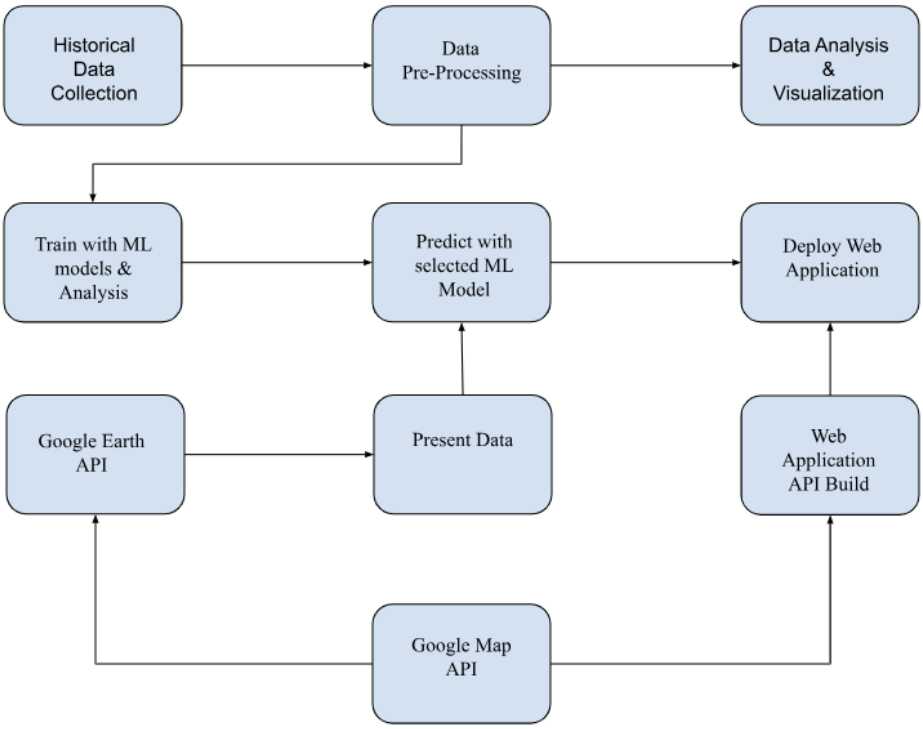


Figure 3.1: System Overview

Chapter 4

Methodology

The researchers have been doing lots of types of explorations for the last ten years for the proper and consistent implementation of Artificial Intelligence for landslide prediction. Promising outputs were obtained and those researches influenced the landslide prediction current system. The two technologies namely, Machine Learning and Artificial Intelligence are also applied in the field of landslide prediction. Previously investigated results and conclusions were briefed in the last chapter. The present study approach and research methodology were incorporated with shortcomings that have also been identified. The methodology adopted to reach the conclusions in this work is briefly discussed in this chapter.

4.0.1 Introduction

The researchers have been doing lots of types of explorations for the last ten years for the proper and consistent implementation of Artificial Intelligence for landslide prediction. Promising outputs were obtained and those researches influenced the landslide prediction current system. The two technologies namely, Machine Learning and Artificial Intelligence are also applied in the field of landslide prediction. Previously investigated results and conclusions were briefed in the last chapter. The present study approach and research methodology were incorporated with shortcomings that have also been identified. The methodology adopted to reach the conclusions in this work is briefly discussed in this chapter.

4.0.2 Methodology Overview

Historical data was first collected. We preprocessed the data and divided it into two parts: training and testing. Historical data was examined, and visualizations represented the contributions of various features in landslide events. Preprocessed data that had been divided for use in training was again used to train various machine learning models.
Google Map API was used in fetching the coordinates of a location, and the fetched data was sent to Google Earth Engine to extract the value of the feature from different satellite images. The value of the feature Land Cover was extracted from MODIS/006/MCD12Q1 Dataset. The feature values of Elevation, Slope, and Aspect were fetched from USGS/SRTMGL1_003 Dataset. NDVI feature value was calculated from LANDSAT/LC08/C01/T1_TOA3 Dataset. Flow Accumulation Area was taken from MERIT/Hydro/v1_0_1 Dataset. Using this, TWI and SPI were

calculated with the help of the Slope value which was extracted in the previous steps. Rainfall data was obtained from JAXA Global Rainfall Watch. NDVI, TWI, and SPI were measured with the following formula:

$$\text{NDVI} = \frac{\text{NIR(BAND4)} - \text{RED(BAND3)}}{\text{NIR(BAND4)} + \text{RED(BAND3)}} \tag{4.1}$$

$$\text{SPI} = A_s \tan \beta \tag{4.2}$$

$$\text{TWI} = \ln \left(\frac{\alpha}{\tan \beta} \right) \tag{4.3}$$

Here,

- A_s = Flow Accumulation Area (m^2)
- β = The local slope gradient (radian)
- α = Cumulative upslope area draining through a point (m^2)

One model was selected to best fit the performance of existing models for the prediction with current data, derived from various satellite images through Google Earth Engine.
The web application was done by using frontend and backend technologies. As Frontend, we used HTML, CSS, and Google Maps API, while as Backend, we used Flask, which is based on Python. In the aim of having the best prediction performance, we deployed the chosen model to the web application.

4.0.3 Method Implementation

The steps in the implementation process have been put in this section in a sequential manner, as shown in figure of methodology overview.

Data Collection

The most crucial step in resolving any machine learning challenge is data collection. However, it is a big hurdle for many researchers and computer scientists. The process of data analysis, including data collection, data labeling, and upgrading existing data or models, is usually very time-consuming. We have faced a huge problem collecting the dataset due to unavailability and insufficiency of the data. Other than 4 major landslides dating back to 2007, there was no official inventory of all landslides that took place in Rangamati no matter how back to date. However, there's been a local inventory (1) of landslides that took place in Rangamati, but the locations were not listed properly. Since no proper inventory of landslides exists for the area, we had to use a third-party dataset (2) published in 2020, based on the local inventory stated above.

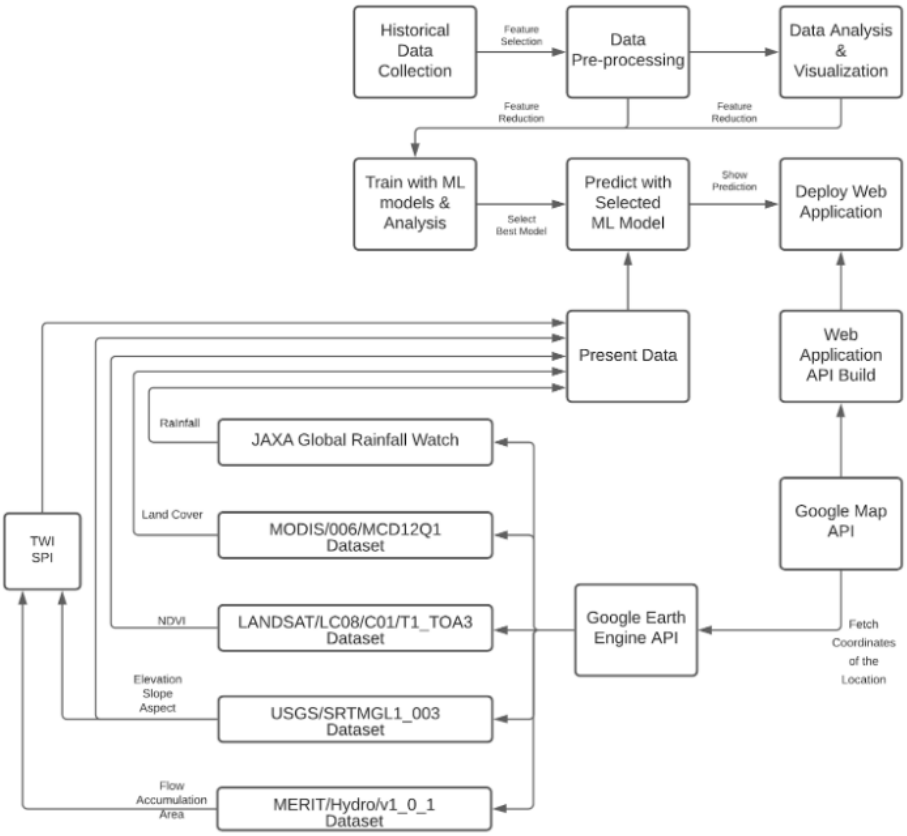


Figure 4.1: Overview of Methodology

Feature Selection

One of the dimensionality reduction techniques is feature selection, since it reduces the number of input variables needed to make a prediction. It can be applied to try and reduce the cost of computation for simulations and sometimes even increase accuracy by providing fewer input variables. The determination of the correlation of each input variable to the target variable is done using statistics, and then the highly correlated input variables are chosen. Since the choice of statistical metrics is based on the type of data—be it for input or output variables—such techniques could be simple yet very effective. Generally, feature selection has three important objectives: it reduces training time, avoids the dimensionality curse, and makes the machine learning model simpler. We have considered 8 features in order to predict the probability of landslide:

- Elevation
- Slope
- Aspect
- Land Cover
- NDVI (Normalized Difference Vegetation Index)

- TWI (Topographic Wetness Index)
- SPI (Stream Power Index)
- Rainfall

We have plotted a correlation matrix for the correlation of primary features in order to know which are the features that are significant and which are the insignificant features. The model can be made more efficient by removing the features that are not necessary for prediction. But neither of those features exist in our work, so no functionality is lost or changed.

Data Preprocessing

The step in any Machine Learning process whereby the given data is translated, or encoded, to make it an easy job for the machine to process is referred to as data preprocessing. In other words, the algorithm can now easily understand the properties of the results. The raw data cannot be understood by machines. Moreover, the processing of string or text data does not enable prediction. To classify, algorithms need data to be projected in a graph or plane. Thus, the raw data obtained has to be transformed into numerical data before machine learning algorithms can be used to arrive at the intended outcome. One important part of traditional data preprocessing steps is data cleaning. Much of the portions of the data may be useless or incomplete. To cope with this part, data cleaning is done. It includes dealing with data which are imperfect, noisy and so on. Nevertheless, all of our data that we gathered were numeric and did not contain any gaps. Therefore, in this case, there is no cleaning of data required. We normalized our dataset to improve the performance since the values were ranging from many ranges. There is no decrease in dimensionality.

Data Analysis

Data analysis is a very critical stage of the data collection process. Data analysis helps data engineers identify the final set of features that are to be used in the machine learning algorithms in predicting the outcomes. During this process, the relation of the features becomes clear. It further helps a data engineer identify the attributes or factors that are irrelevant and can thus be eliminated. The result of the analysis helps to interpret the story that the data is trying to tell and to determine its relevance.

Model Evaluation and Selection

With machine learning libraries like scikit-learn, it is trivial to fit a long list of machine learning models on a predictive modeling dataset. Consequently, one of the most challenging elements in applied machine learning is selecting which model to use on a given problem from a large suite of possible choices. Model selection is the process involved in choosing between different types of models and also between models of the same type but differing in model hyperparameters. The phase where one will be in a position to estimate the generalization error is what is used to refer to model evaluation. A good machine learning model, aside from doing well on seen

data, will do well with unseen data as well. This will make us fairly confident that the performance of a model won't degrade when it is subjected to new data before deploying to production. Model Selection For this work, a number of machine learning models are trained using the collected preprocessed data. Some of the machine learning algorithms, which have been trained and tested, are given below:

- K-Nearest Neighbors (KNN)
- Decision Trees (DT)
- Support Vector Machine (SVM)
- Linear Regression (LR)
- Random Forest (RF)
- Bayesian Regression (BR)
- AdaBoost (AB)
- Gradient Tree Boosting (GTB)
- Neural Network (NN)
- Soft Voting Ensemble Model (Hybrid Model)

Soft Voting Ensemble Model (Hybrid Model) is selected and implemented by using the Web Application after the comparison of the machine learning algorithms in terms of their performance.

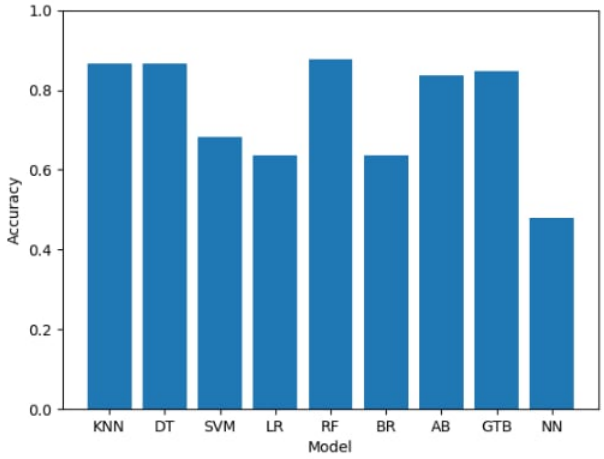


Figure 4.2: Model Accuracy

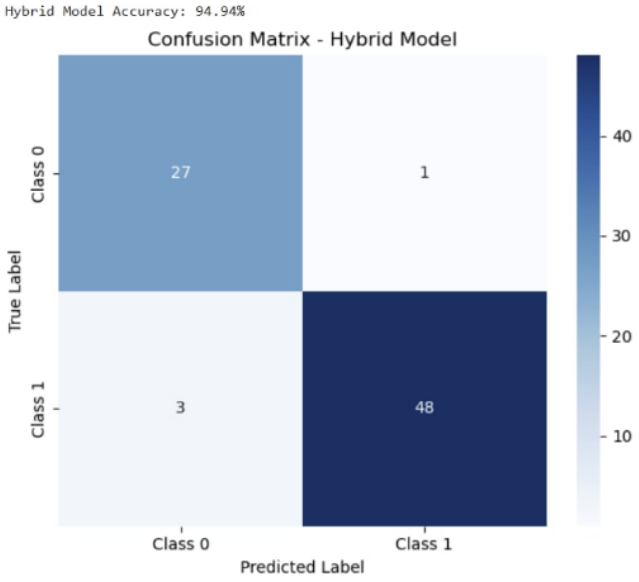


Figure 4.3: Performance evaluation of the Soft Voting Ensemble Model (Hybrid Model) using Confusion Matrix

Web Application Development

Frontend Development The art of writing HTML, CSS, and JavaScript code for a website or a web application in such a way that users may have the ability to access and communicate directly with it is called front-end web development. It is also referred to as client-side development. Hence, the biggest challenge in the construction of a website frontend lies in the fact that methods and techniques always go out of trends, therefore developers need to know and grasp how advancements happen. The goal of a website design should be that whenever people come to the site, they see well-structured and readable content. The challenge of frontend construction has to do with the fact that approaches and techniques for building the frontend of a website change every so often. The developer, therefore, needs an understanding of ways in which the field progresses, regularly. The purpose of web design is to make sure that when visitors come to the website, they obtain easily readable content. CSS beautifies the interface.

Backend Development Backend architecture is server-side development and deals mostly with how the web works. Your key responsibilities will include making changes, upgrades, and managing the functionality of the site. This type of web development usually consists of a server, an application, and a database. The code written by backend developers enables the browser to interact with the database. A backend developer is responsible for things that aren't visible to the naked eye, like databases and servers. Also known as backend developers, Programmers or web developers: The backend of the Web Application, in some cases, is developed using Python programming and the Flask microservice. A selected machine learning model, which has been chosen based on performance, is trained and saved as a pickle file. Thereafter, the model will be loaded into the backend code base. In the backend, two APIs will be developed: the main page API and the predict API. When anyone clicks on the map with the mouse, the information will be retrieved via an HTML form on the client side of the Web Application and then fetched via API to the backend. When the mouse is clicked, the coordinates of that location will pass through the backend and extract the features from the satellite images, and then pass through the trained model to get the prediction result. The result will then be rendered on the application frontend.

4.1 Conclusion

The aim of this study is to present a machine learning based approach to successfully predict the occurrence of landslides in Rangamati hill tracts. The section describes the system used for the achievement of the work's objective in detail. First, we started the process with data collection . A set of features is selected for the procedure to be achieved. Further, the analysis and visualization of data are carried out to better understand the importance of features and mutual correlations of the features. With this final selected feature dataset, it trains different machine learning models. Based on the performance of the models, one model is selected to perform the prediction. A web application will be developed, the model will be deployed in the web application for usability and will successfully enable real time monitoring. Which will be helpful in warning the people beforehand and minimizing the loss.

Chapter 5

Result Analysis & Discussion

5.1 Overview

This is where we get to the gist of our research, giving you the most important findings in summary and explaining them to you in simple English. We have looked at the data carefully, crunched the numbers, and now we are ready to tell you what it all amounts to. Here, we give you the results of our research, supported by numerical data, and provide some insights and recommendations on the basis of what we discovered. This chapter will assist other researchers with what to do and what can be done when applying the same method.

In the last chapter, we took you through the bread and butter of our method—how we gathered the data, selected the most applicable features, and interpreted it all. We also tried out some machine learning algorithms to determine which worked best. Now, in this chapter, we're laying out the results of that process, highlighting the main takeaways and lessons learned. We've included some visuals, like graphs and charts, to make the data easier to grasp, along with explanations to put everything into context.

5.2 About the Dataset

Our data consist of 392 landslide events, half and half in terms of positive (landslide) and negative (no landslide) cases. We have eight most pertinent features with significant landslide impact to estimate their probability.
Figure 5.1 illustrates the distribution among the two classes (landslide or no landslide) of the 392 records. The data are well balanced with half the number of cases in each set.

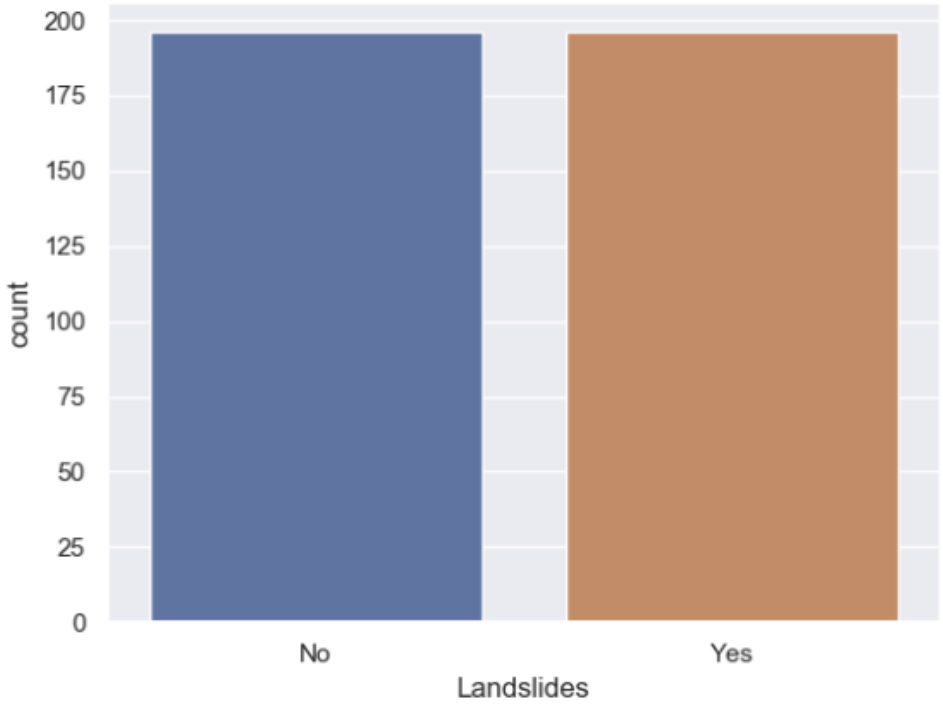


Figure 5.1: No. of Records

Figure 5.2 depicts the interdependencies among the features. Slope, Elevation, NDVI (vegetation measurement), TWI (topographic index), and Rainfall seem to be most related to landslide prediction. But NDVI won't likely vary much over time at any spot, and TWI depends on Slope, so they're not quite as independent. That leaves Slope, Elevation, and Rainfall as the heavy hitters for landslide prediction.

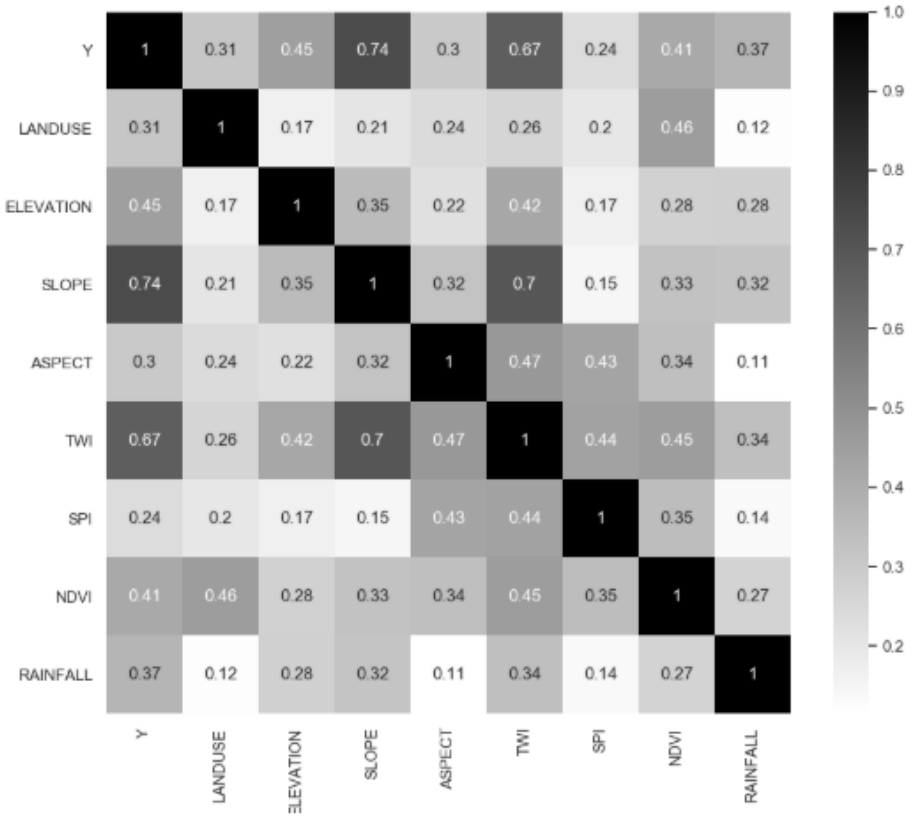


Figure 5.2: Correlation Matrix

Figure 5.3 shows that the feature data is dispersed. Almost all of the features have a dense cluster of values, while others, such as Rainfall, are much more variable, showing its role in causing landslides.

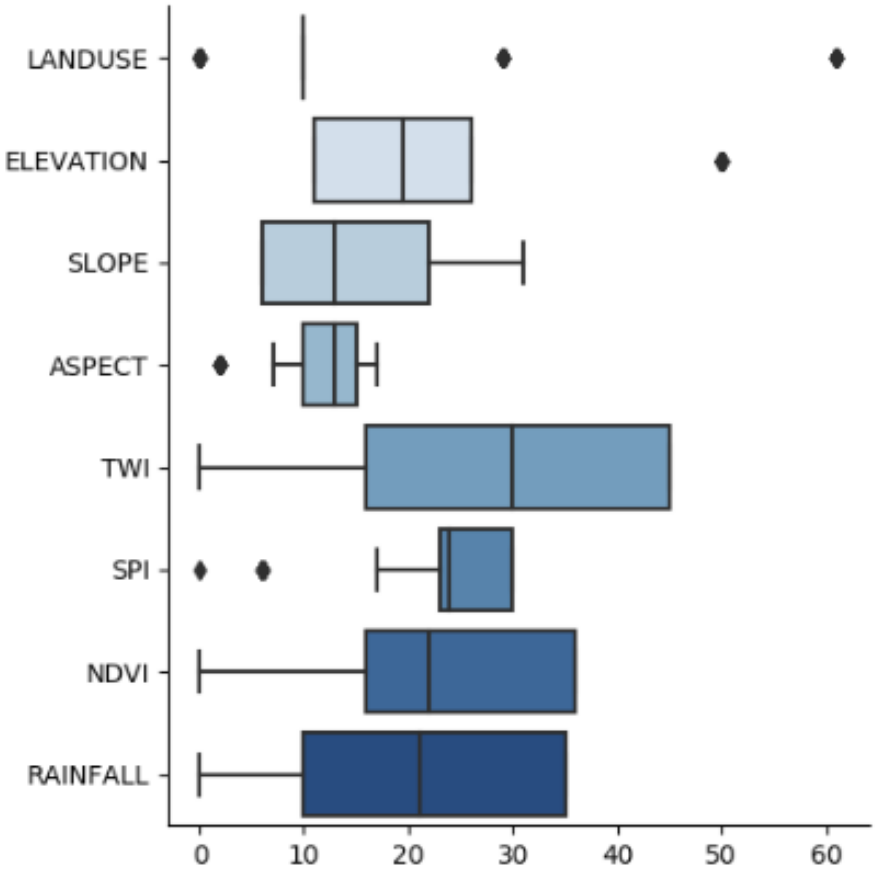


Figure 5.3: Features Data Distribution

Figure 5.4 brings it all together, showing the combined impact of Rainfall, Slope, and Elevation. The takeaway? Steep slopes paired with heavy rainfall significantly increase the chances of a landslide. High elevation amps up the risk even more when combined with steep slopes and intense rain.

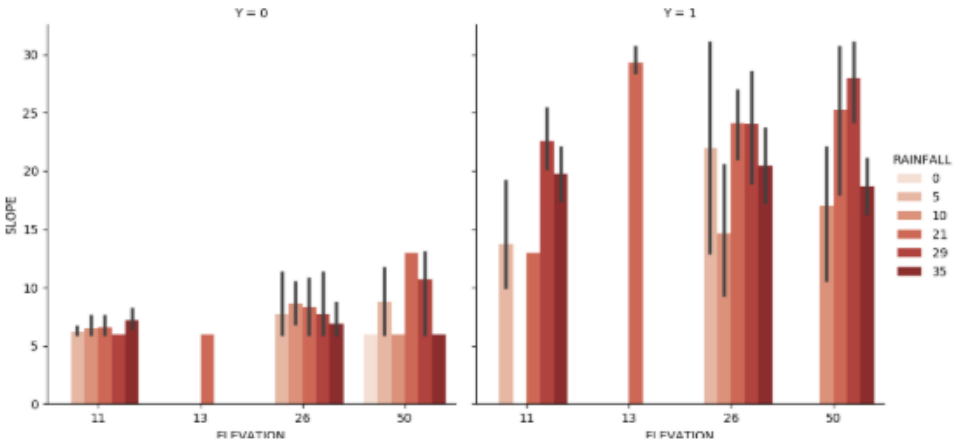


Figure 5.4: Impact of Rainfall, Slope and Elevation

5.3 Impact Analysis

Machine learning is revolutionizing the world in amazing ways, and its uses are becoming a more and more common part of our daily lives. Our project is no exception—it can genuinely make a difference. Here's how:

1. **Identifying Landslide Risk:**: Anyone can easily identify the risk of a landslide in any location with our web application. It's a straightforward tool that places important information at your fingertips.

2. **Saving Lives through Early Warnings:** The app can send warnings to alert people of an impending landslide, saving lives and property before they're lost.

3. **Mapping High-Risk Zones:** By setting a threshold of risk, our system can identify areas with the most susceptible landslide risk, making it simpler to plan and prepare.

5.4 Impact of Rainfall in Rangamati Landslides

We looked specifically at five of the largest Rangamati landslides to determine what had caused them. Four of them were the result of excessive rainfall, we were told by our sources, and one was caused by excavation. We dug deeper by analyzing rainfall patterns for the month prior to each slide. The findings were appalling and corroborated exactly as described by the sources.
For the four heavy rain-induced landslides, we observed that they were all triggered by at least 15 days of heavy rainfall exceeding 200 mm cumulatively. For the excavation-induced landslide, however, it was just 40 mm of rain during the same time frame, so it would have been too unlikely that rain would have played a big role there.

Date	Location		Cause	Rainfall Recorded Last 15 days
	Latitude	Longitude		
11/06/2007	22.64	92.15	Heavy Rainfall	221.91 mm
16/03/2017	22.65	92.15	Digging	39.86 mm
12/6/2017	22.59	92.16	Heavy Rainfall	702.83 mm
13/06/2017	22.65	92.16	Heavy Rainfall	1120.64 mm
13/06/2017	22.75	92.22	Heavy Rainfall	941.14 mm

Here is the breakdown data, with accompaniment graphs:

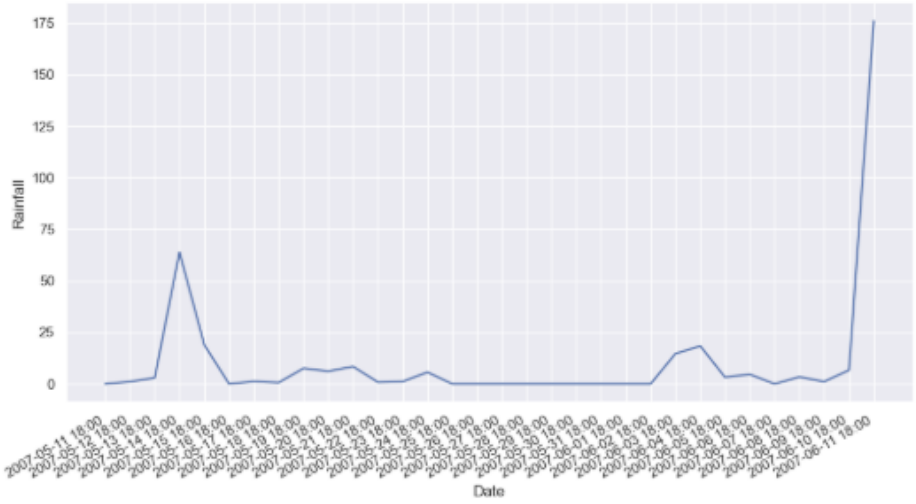


Figure 5.5: Rainfall Record (Day to Day)
11/06/2007
Latitude: 22.64 Longitude: 92.15

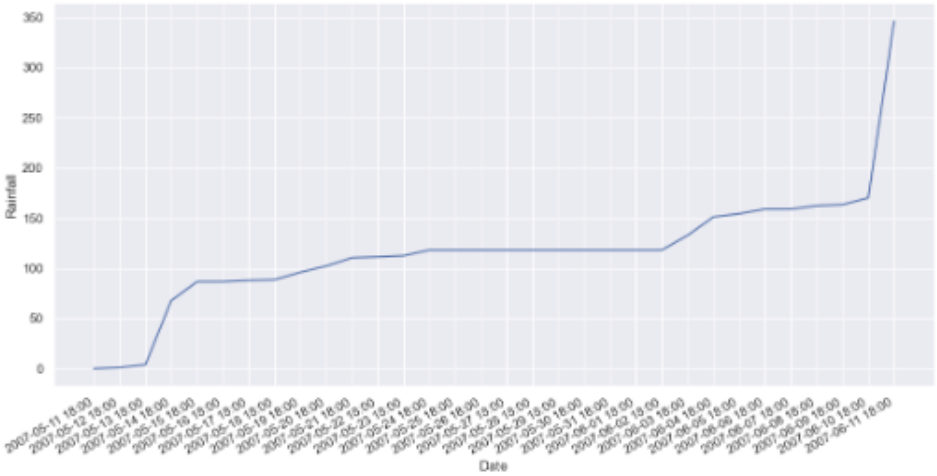


Figure 5.6: Rainfall Record (Cumulative)
11/06/2007
Latitude: 22.64 Longitude: 92.15

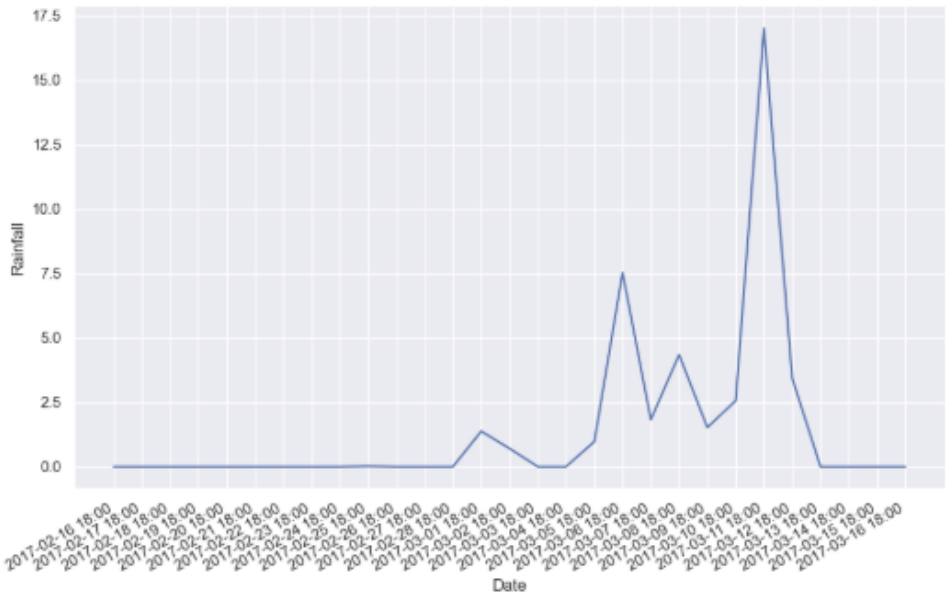


Figure 5.7: Rainfall Record (Day to Day)
16/03/2017
Latitude: 22.65 Longitude: 92.15

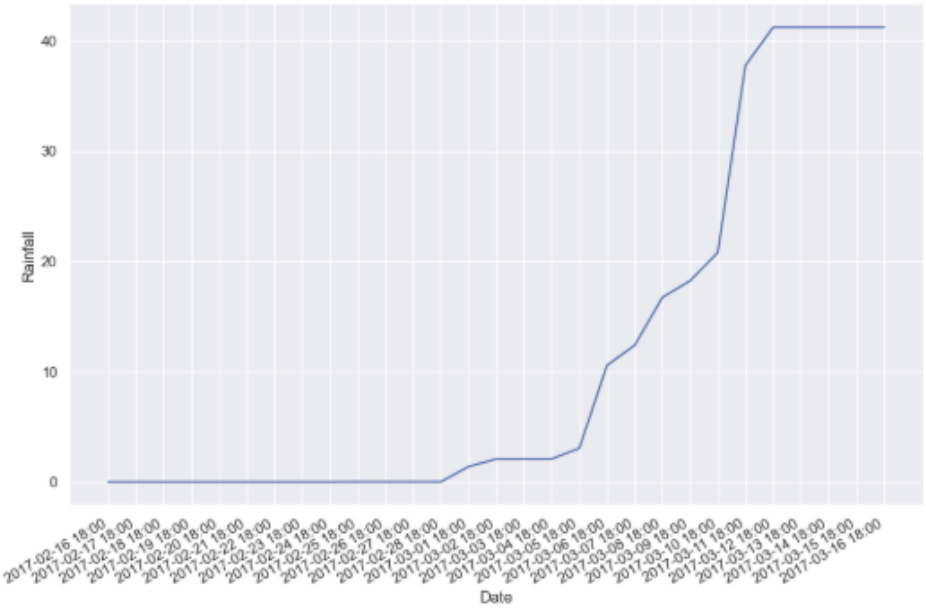


Figure 5.8: Rainfall Record (Cumulative)
16/03/2017
Latitude: 22.65 Longitude: 92.15

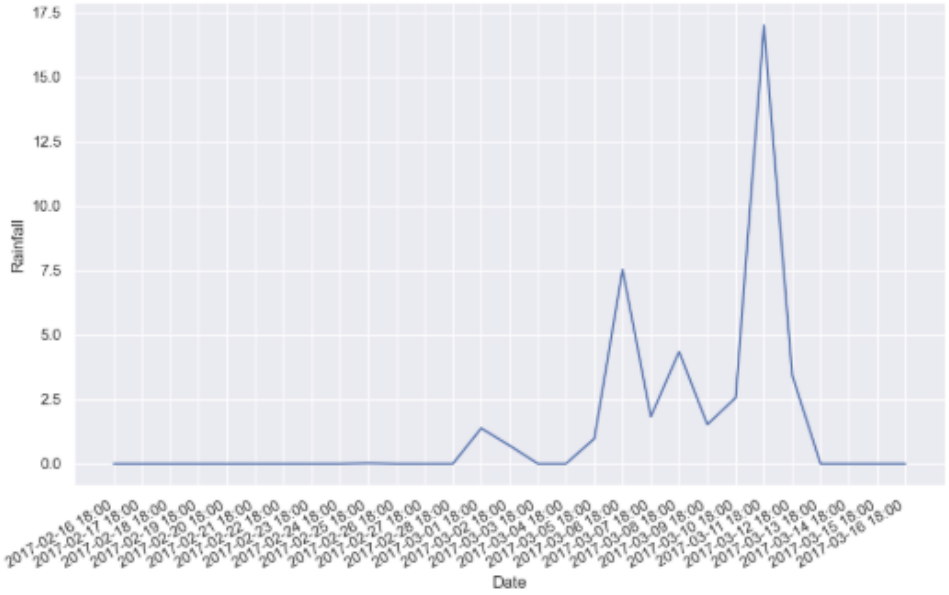


Figure 5.9: Rainfall Record (Day to Day)
12/06/2017
Latitude: 22.59 Longitude: 92.16

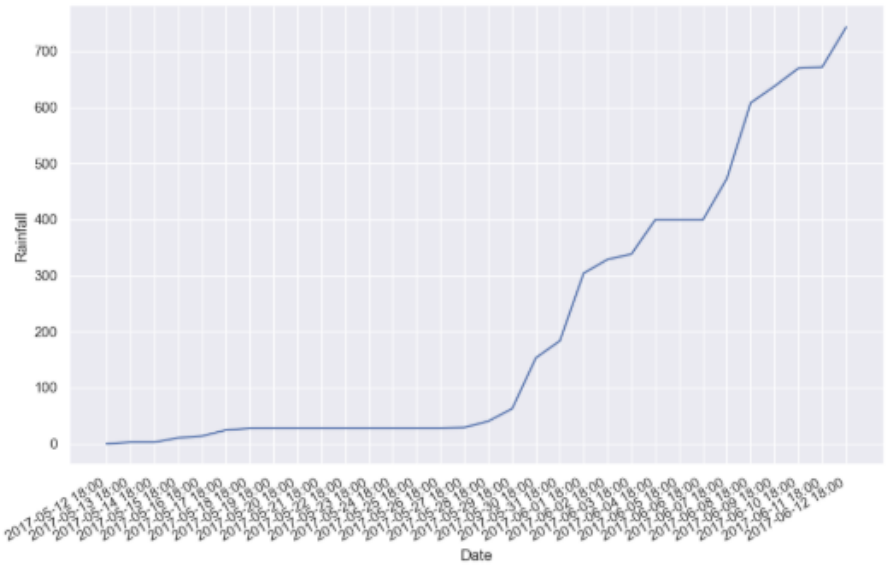


Figure 5.10: Rainfall Record (Cumulative)
12/06/2017
Latitude: 22.59 Longitude: 92.16

Figures 5.9 5.10: During May 12 to June 12, 2017, with location 22.59N, 92.16E, an unbelievable 702.83 mm rain was recorded for the last 15 days. The landslide killed at least 100 people, four of them soldiers, in various locations in Rangamati. The heavy rain was the primary reason.[11]

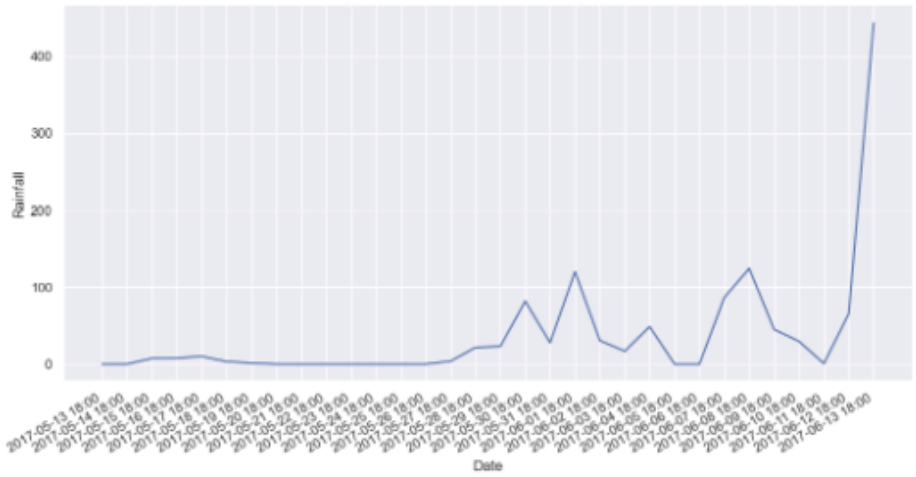


Figure 5.11: Rainfall Record (Day to Day)
13/06/2017
Latitude: 22.65 Longitude: 92.16

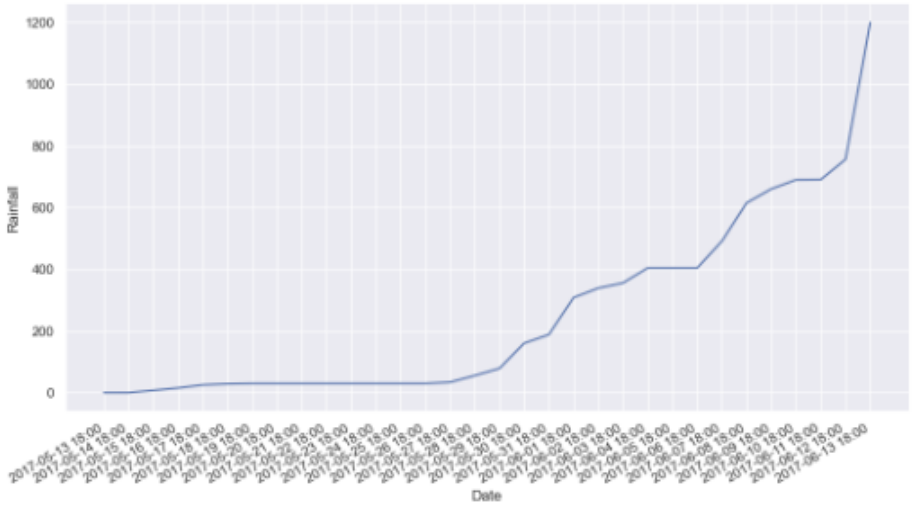


Figure 5.12: Rainfall Record (Cumulative)
13/06/2017
Latitude: 22.65 Longitude: 92.16

Figures 5.11 5.12: On coordinates 22.65N, 92.16E, between May 13 and June 13, 2017, there was another heavy rain that led to yet another landslide that killed at least 100 individuals, including four soldiers, in Rangamati.[11]

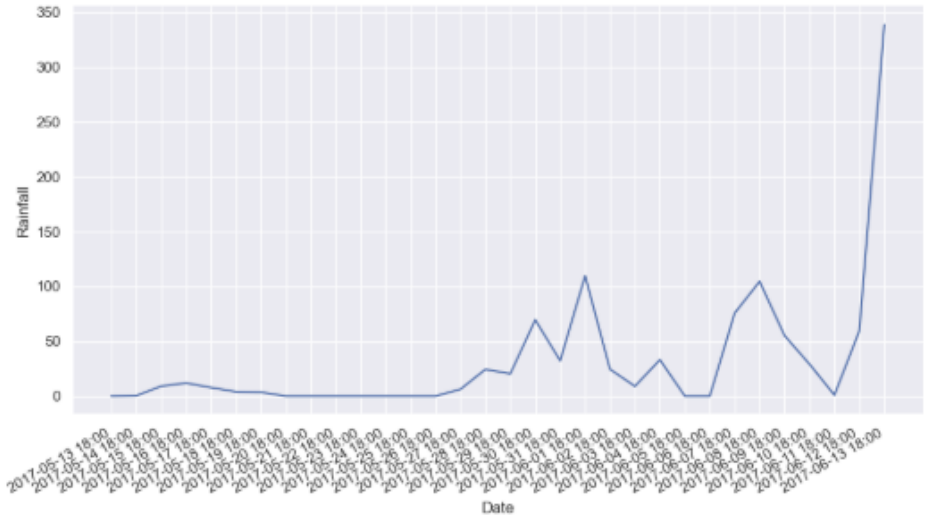


Figure 5.13: Rainfall Record (Day to Day)
13/06/2017
Latitude: 22.75 Longitude: 92.22

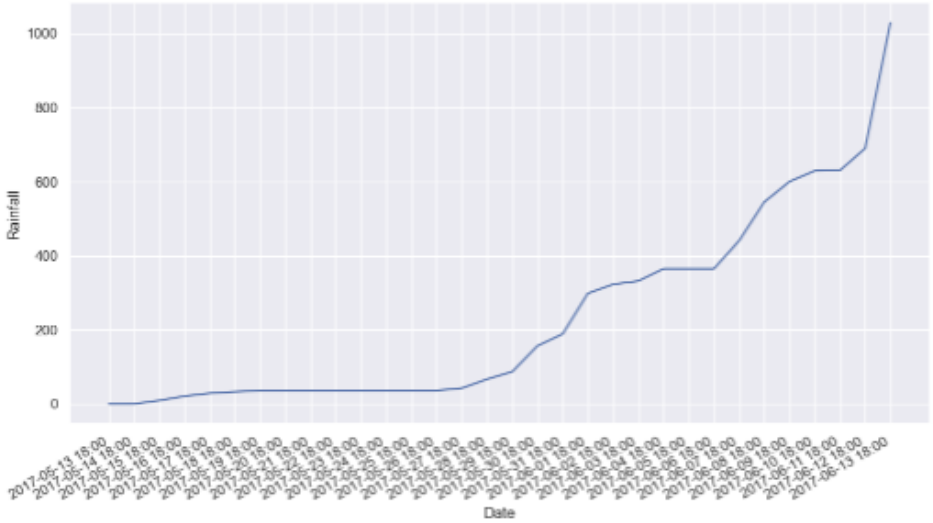


Figure 5.14: Rainfall Record (Cumulative)
13/06/2017
Latitude: 22.75 Longitude: 92.22

Figures 5.13 5.14: At latitude 22.75N, longitude 92.22E, from May 13 to June 13, 2017, an all-time high 941.14 mm of rain pounded the last 15 days. This was followed by yet another fatal landslide that claimed at least 100 lives, four of them soldiers, in Rangamati. Rainfall was a prime suspect.[11]

5.5 Testing How Well Our Models Work

Machine learning model testing is an important process, though, and occasionally it's not straightforward. It does take some working through to get it right. Only using one measure, such as accuracy, can give a model the bad or good by some measures but fall short when counted in others, such as logarithmic loss. Accuracy is one measure of how good a model is for prediction but isn't the complete story.

We tested nine well-known machine learning classifiers along with a hybrid model that we created in our study in order to compare their performance against our perfectly created dataset. Their performance was measured by accuracy, precision, recall, and F1 score—parameter metrics by which we will determine the best-performing model. We divided our dataset into 75% for training the model and 25% for the test run to ensure we obtained credible results. Each of the models was trained on a randomly shuffled segment of the data and then tested on the remaining block to determine how it would perform with new data.

Table 5.1: Performance Evaluation of ML Models

Classifier	Accuracy	Precision	Recall	F1 Score
K-Nearest Neighbors (KNN)	0.87	0.8780	0.8709	0.8670
Decision Trees (DT)	0.87	0.8675	0.8667	0.8670
Support Vector Machine (SVM)	0.68	0.8110	0.6702	0.6374
Linear Regression (LR)	0.64	0.8176	0.7353	0.7084
Random Forest (RF)	0.88	0.8790	0.8790	0.8776
Bayesian Regression (BR)	0.64	0.8092	0.7157	0.6835
AdaBoost (AB)	0.84	0.8381	0.8381	0.8367
Gradient Tree Boosting (GTB)	0.85	0.8495	0.8488	0.8469
Neural Network (NN)	0.48	0.2398	0.5000	0.3241
Soft Voting Ensemble Model (Hybrid Model)	0.94	0.9514	0.9494	0.9497

It can be seen from 5.1 that our Soft Voting Ensemble Model (Hybrid Model) yields the best performance with an accuracy of 0.94 as well as the optimal precision, recall and F1 score.

5.6 Our Web Application

We deployed our model as a web application using Flask, a lightweight web framework, in order to be accessible to our users. Its behavior is as follows:

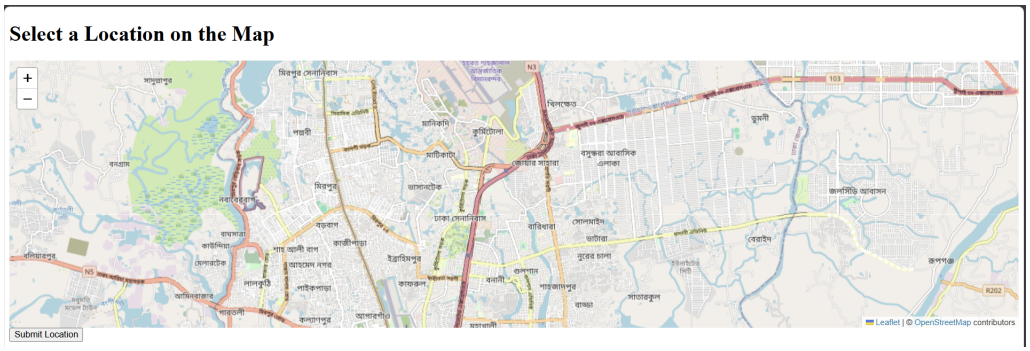


Figure 5.15: Home Page

Figure 5.15: The home page encourages the user to click on a map so you can view the landslide hazard for where you are.

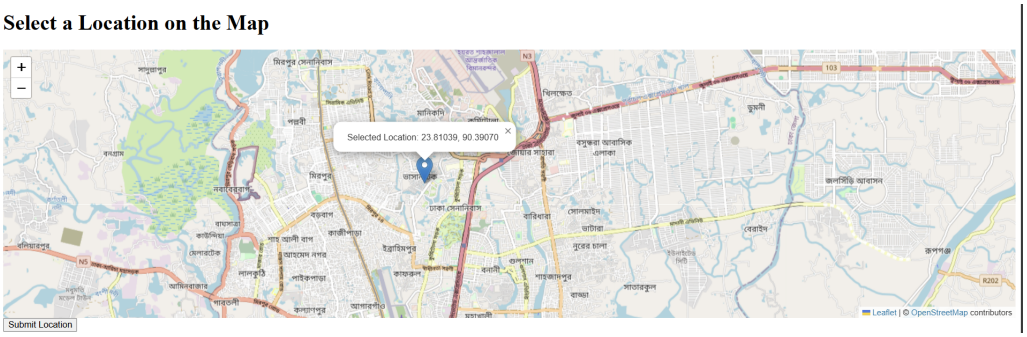


Figure 5.16: Select Location

Figure 5.16: Upon clicking the map, you are shown a loading screen while the app fetches your request.

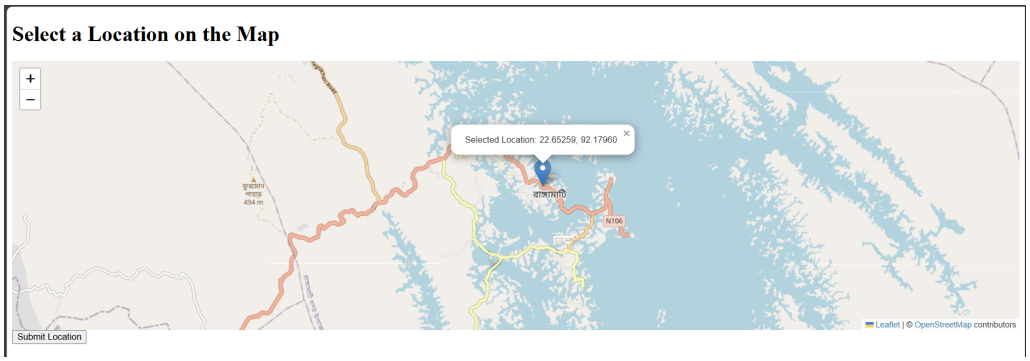
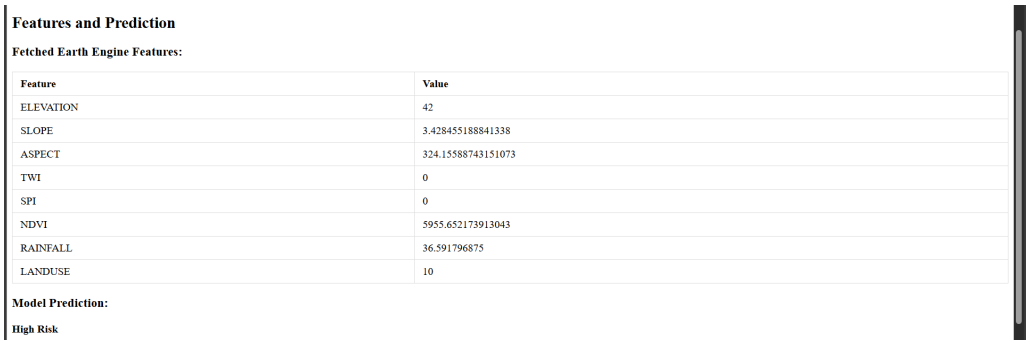


Figure 5.17: A location in Rangamati has been selected



Features and Prediction

Fetched Earth Engine Features:

Feature	Value
ELEVATION	42
SLOPE	3.428455188841338
ASPECT	324.15588743151073
TWI	0
SPI	0
NDVI	5955.652173913043
RAINFALL	36.591796875
LANDUSE	10

Model Prediction:

High Risk

Figure 5.18: High Risk

Figure 5.18: on clicking, the app determines feature values (e.g., slope or rainfall) for the place fetching the data from Google Earth Engine API and shows the landslide risk with values and the risk (Here it shows high risk) when there's a 60% or higher landslide probability

5.7 Conclusion

A model's ability to predict accurately on new, unseen data is what really matters. Our goal was to build something that can reliably forecast landslide risks based on future conditions. Since every machine learning algorithm tackles problems differently depending on the dataset and goal, choosing the right way to measure success is crucial. In this chapter, we have presented our findings, beginning with an overview of the dataset and the most important conclusions that we made based on it. We further presented the testing results of our model and what we were able to achieve by doing so. To put this into perspective, we compared our work with work conducted by other researchers. And lastly, we showcased the web app's interface, created with a focus on making the tool as usable and efficient as it could be. In the next chapter, we shall wrap up our research and abstract.

Chapter 6

Conclusion & Future Work

The hill regions of Bangladesh have in recent years suffered greatly from landslides, and this is evidence of how imminent the threat is to cities and to citizens of the cities. Landslides can cause great harm and can have economically and socially devastating effects, especially because such cities are growing at a rapid rate and contribute largely to the nation's economy. That's why it's so important to develop a system for managing landslide risks and to better understand the scope, severity, and potential consequences for communities living in vulnerable areas.
Our approach is all about learning from the past to prepare for the future. By using machine learning, we've created a system that predicts the likelihood of landslides in real-time and issues early warnings to help keep people safe.

In Chapter 1, we provided an overview of the project in general and presented the main ideas. We also touched on the issues and how we managed to overcome them, what necessitated the work and the importance.

Then we learned about how machine learning has already been applied in other domains, such as medicine to detect illness. We reviewed current research, the approach, data gathering, constraints, and algorithms employed, and a brief overview of how conventional machine learning algorithms are used in our project.

Chapter 4 outlines step by step the way we moved from data gathering to releasing our model, describing everything to reveal how we managed to achieve our objective.

We shared our findings in Chapter 5. We initially described the dataset and then emphasized the most important discoveries made with our analysis, such as the very important role rainfall has in initiating landslides. We also shared our model validation and contrasted our results with those of other scholars. And, we noted the clean user interface of our web application, which we made simple and assistive.

One of the main takeaways is that landslide prediction in a specific region needs historical data regarding slides in the region. Landslides are locally triggered by weather and terrain, so it may not be possible to use data from one region for another. Geospatial conditions like slope and elevation are generally the primary reasons for such activities.

6.1 What's Next

There is also plenty of scope for expanding on this work. Some ideas for the future include:

1. Build a Global Landslide Database: It would enhance the credibility and soundness of our dataset if we also had a global, comprehensive database of landslides.

2. Make an App that will be more User-Friendly: We'd love to make an app other than this web application that will be more intuitive and user-friendly.

3. Interfacing with Local Forecasting Systems: Interfacing the app with local weather forecasting systems would facilitate provision of timely early warnings.

4. Scaling Up Beyond Rangamati: We intend to scale this up for use throughout Bangladesh and, eventually, other nations, widening coverage.

Through further development and expansion of this effort, our aim is to make communities safer and more resilient to landslides wherever they occur.

Bibliography

[1] E. Fix, *Discriminatory analysis: Nonparametric discrimination, consistency properties*. USAF School of Aviation Medicine, 1985.

[2] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[3] T. K. Ho, "Random decision forests," in *Proceedings of 3rd International Conference on Document Analysis and Recognition*, IEEE, vol. 1, 1995, pp. 278–282.

[4] Y. University, *Linear regression*, http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm, 1997.

[5] X. Wu, V. Kumar, J. R. Quinlan, *et al.*, "Top 10 algorithms in data mining," *Knowledge and Information Systems*, vol. 14, no. 1, pp. 1–37, 2008.

[6] J. Tingyao and W. Dinglong, "A landslide stability calculation method based on bayesian network," in *2013 2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation (IMSNA)*, IEEE, 2013, pp. 905–908.

[7] O. Korup and A. Stolle, "Landslide prediction from machine learning," *Geology Today*, vol. 30, no. 1, pp. 26–33, 2014.

[8] F. Ren and X. Wu, "Landslide displacement prediction affected by the periodic precipitation, reservoir level and groundwater level fluctuations," *International Journal of u-and e-Service, Science and Technology*, vol. 7, no. 5, pp. 235–250, 2014.

[9] *3 killed in rangamati landslide*, https://www.dhakatribune.com/bangladesh/nation/2017/03/16/3-killed-rangamati-landslide, Accessed on 31 Mar 2017, Mar. 2017.

[10] K. Agrawal, Y. Baweja, D. Dwivedi, *et al.*, "A comparison of class imbalance techniques for real-world landslide predictions," in *2017 International Conference on Machine Learning and Data Science (MLDS)*, IEEE, 2017, pp. 1–8.

[11] T. D. S. O. Report. "Landslide in bangladesh: Death toll jumps to 133." Accessed on pp. 33–35. (Jun. 2017), [Online]. Available: https://www.thedailystar.net/country/16-killed-bandarban-rangamati-landslides-1419571.

[12] W. Koehrsen, *Introduction to bayesian linear regression*, https://towardsdatascience.com/introduction-to-bayesian-linear-regression-666e60791ea7, 2018.

[13] A. Shirzadi, K. Soliamani, M. Habibnejhad, *et al.*, "Novel gis-based machine learning algorithms for shallow landslide susceptibility mapping," *Sensors*, vol. 18, no. 11, p. 3777, 2018.

[14] L. Xiao, Y. Zhang, and G. Peng, "Landslide susceptibility assessment using integrated deep learning algorithm along the china-nepal highway," *Sensors*, vol. 18, no. 12, p. 4436, 2018.

[15] Y. Wang, X. Wu, Z. Chen, F. Ren, L. Feng, and Q. Du, "Optimizing the predictive ability of machine learning methods for landslide susceptibility mapping using smote for lishui city in zhejiang province, china," *International Journal of Environmental Research and Public Health*, vol. 16, no. 3, p. 368, 2019.

[16] J. Chen, *Neural network definition*, https://www.investopedia.com/terms/n/neuralnetwork.asp, 2020.

[17] Displayr, *Gradient boosting explained - the coolest kid on the machine learning block*, https://www.displayr.com/gradient-boosting-the-coolest-kid-on-the-machine-learning-block/, 2020.

[18] *2007 chittagong mudslides*, https://en.wikipedia.org/wiki/2007_Chittagong_mudslides, Accessed on 31 May 2021, May 2021.

[19] CIMSS, *Satellite images*, https://cimss.ssec.wisc.edu/oakfield/sat.htm, 2021.

[20] G. E. Engine, *Mcd12q1.006 modis land cover type yearly global 500m*, https://developers.google.com/earth-engine/datasets/catalog/MODIS_006_MCD12Q1, 2021.

[21] G. E. Engine, *Merit hydro: Global hydrography datasets*, https://developers.google.com/earth-engine/datasets/catalog/MERIT_Hydro_v1_0_1, 2021.

[22] G. E. Engine, *Nasa srtm digital elevation 30m*, https://developers.google.com/earth-engine/datasets/catalog/USGS_SRTMGL1_003, 2021.

[23] G. E. Engine, *Usgs landsat 8 collection 1 tier 1 toa reflectance*, https://developers.google.com/earth-engine/datasets/catalog/LANDSAT_LC08_C01_T1_TOA, 2021.

[24] Google, *Google earth engine api*, https://earthengine.google.com/, 2021.

[25] JAXA, *Jaxa climate rainfall watch*, https://www.space4water.org/s4w/web/softwaretoolweb-app/jaxa-climate-rainfall-watch, 2021.

[26] G. Learning, *Adaboost algorithm: Boosting algorithm in machine learning*, https://www.mygreatlearning.com/blog/adaboost-algorithm/, 2021.

[27] A. l. a. g. e. e. M.Altaweel, *A look at google earth engine*, https://www.gislounge.com/a-look-at-google-earth-engine/, 2021.

[28] Mozilla, *Css: Cascading style sheets*, https://developer.mozilla.org/en-US/docs/Web/CSS, 2021.

[29] Mozilla, *Html: Hypertext markup language*, https://developer.mozilla.org/en-US/docs/Web/HTML, 2021.

[30] Pallets, *Flask: A python microframework*, https://flask.palletsprojects.com/, 2021.