Bachelor of Science in Computer Science & Engineering

Landslide Prediction using Machine Learning Algorithm of Rangamati Hill Tracts

by

Ricko Shaha

ID: 1504008

Department of Computer Science & Engineering

Chittagong University of Engineering & Technology (CUET)

Chattogram-4349, Bangladesh.

May, 2021

Landslide Prediction using Machine Learning Algorithm of Rangamati Hill Tracts

Submitted in partial fulfilment of the requirements for
Degree of Bachelor of Science
in Computer Science & Engineering

by

Ricko Shaha
ID: 1504008

Supervised by

Dr. Asaduzzaman
Professor
Department of Computer Science & Engineering

Chittagong University of Engineering & Technology (CUET)
Chattogram-4349, Bangladesh.

The thesis titled '**Landslide Prediction using Machine Learning Algorithm of Rangamati Hill Tracts'** submitted by ID: 1504008, Session 2019-2020 has been accepted as satisfactory in fulfilment of the requirement for the degree of Bachelor of Science in Computer Science & Engineering to be awarded by the Chittagong University of Engineering & Technology (CUET).

Board of Examiners

Chairman

Dr. Asaduzzaman
Professor
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

Member (Ex-Officio)

Dr. Md. Mokammel Haque
Professor & Head
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

Member (External)

Dr. Md. Mokammel Haque
Professor
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

Declaration of Originality

This is to certify that I am the sole author of this thesis and that neither any part of this thesis nor the whole of the thesis has been submitted for a degree to any other institution.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. I am also aware that if any infringement of anyone's copyright is found, whether intentional or otherwise, I may be subject to legal and disciplinary action determined by Dept. of CSE, CUET.

I hereby assign every rights in the copyright of this thesis work to Dept. of CSE, CUET, who shall be the owner of the copyright of this work and any reproduction or use in any form or by any means whatsoever is prohibited without the consent of Dept. of CSE, CUET.

Signature of the candidate

Date:

Acknowledgements

I am grateful to Almighty for providing me with the opportunity to finish my thesis and pursue my B.Sc. Engineering degree. Dr. Asaduzzaman, Professor, Department of Computer Science and Engineering, Chittagong University of Engineering and Technology supervised me throughout the process and I am grateful for his guidance, constructive criticism, and unwavering perseverance as the thesis progressed. His insight helped me to solve numerous problem at a glance. He helped me providing books, research papers, took personal sessions to clarify many topics related to the thesis and helped me giving useful advices.

I would also like to convey my gratitude to Debasish Roy Raja, Assistant Professor, Department Of Urban and Regional Planning, Chittagong University of Engineering and Technology for his kind advice. I'd just like to express my gratitude to all of my teachers for their invaluable support over the four years of my learning journey. Finally, I'd like to thank my friends, seniors and department staffs for their helpful suggestions and assistance in completing the project.

Abstract

Landslides are very frequent natural hazards in hill tracts, causing serious damages to human lives and economy. Unlike other major catastrophic events such as earthquakes and floods, landslides rarely have a remarkable negative influence on development projects. It is very difficult to perfectly predict the mechanism of landslides, as they are controlled by many triggering factors. For the past decade, a lot of research has been conducted to employ the ability of machine learning to predict landslide. The aim of this project is to predict the probability of landslide in Rangamati Hill Tracts. In this study a set of 8 features were selected to train the models and predicting landslides. The collected data was analyzed through data count, correlation matrix and distributions of feature data from where we get to a conclusion that three features were mainly causing the trouble. We also analyzed 5 major landslides Rainfall data from where we found out that Rainfall is mainly responsible for the landslides event in Rangamati. The dataset is used to put various machine learning algorithms to the test and analyze several cases and visualizations. The evaluation of models suggested that Random Forest(RF) performs best with an accuracy of 0.88 along with high precision, recall and f1 score.

Keywords— Landslide, Machine Learning, Prediction, Rangamati

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Landslide is a generic term, describing both the formation and processes resulting the displacement of hillslope forming materials- soil, debris, rock driven by the force of gravity, which is also often aided by water. [1] Landslides been very frequent natural hazards in hill tracts, causing serious damages to human lives and economy.[2]

Landslides are becoming a very frequent disaster in several Asian countries. Unlike other major catastrophic events such as earthquakes and floods, landslides rarely have a remarkable negative influence on development projects and the region affected is usually small. As a result, landslide problems have received less attention in many Asian countries along with Bangladesh.

Predicting where and when landslides are likely to take place in a particular region of interest remains a crucial challenge in natural hazards research and alleviation.[1] In past, enormous efforts have been made to predict landslides for efficient hazard management. It is very difficult to perfectly predict the mechanism of landslides, as they are controlled by many triggering factors.[2]

Our intention is to predict the probability of landslide in Rangamati Hill Tracts to generate early warnings to minimize the loss of lives and assets. Real time monitoring is one of prime concern of this project.

This chapter overviews the proposed system to predict probability of landslide of

Rangamati Hill Tracts and challenges faced upon completion of the work. The chapter also summarizes the motivation and contribution of the work to the Landslide Prediction systems.

1.1.1 Objective

1. Collecting historical dataset as well as real time data through satellite image, weather api to predict real time landslide probability of a certain location.
2. Portraying proper visualization of the real time events.
3. Deploying web application for users monitoring and warning.

1.2 Framework/Design Overview

Data is required to feed into the model of Machine Learning Systems. Traditional machine learning models are sensitive to feature selection and the way data is preprocessed. Hence, feature selection is an important step before training the data.

At first historical data was collected. We preprocesssed the data, split it down to train and test set. Historical data was analyzed and visualization portrayed us how features were involved in landslide events. Again with the preprocessed data that was splitted for training purpose was used to train different machine learning models. A model was selected based on the performance of the existing models to predict with the present data which was extracted from different satellite image using Google Earth Engine. A web application was developed using frontend and backend technologies. The selected model was deployed to the web app with ambition of getting the best prediction result.

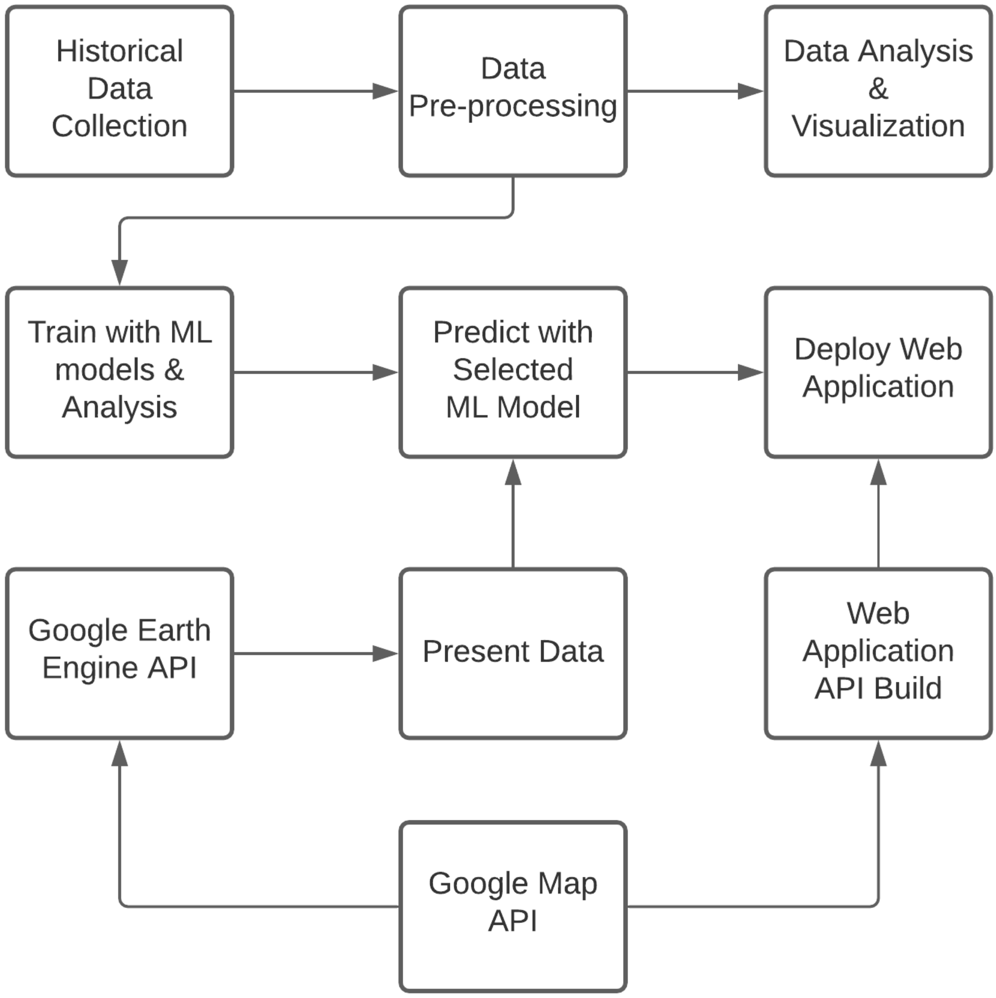

Figure 1.1: System Overview

1.3 Difficulties

There were a several difficulties during the work. The first major problem was unavailability and insufficiency of data. There was no official inventory of all landslides that took place in Rangamati no matter how back to date, other than 4 major landslides date backs to 2007. However, there's been a local inventory of landslides that took place in Rangamati, but the place that landslide took was not listed properly. Again, there were some factors that we couldn't take in account of our consideration even though these factors influences landslide immensely.

1.4 Applications

The proposed system we are considering has got a number of applications.

1. It predicts the real time landslide probability of a certain location.
2. It marks the location of landslide with a probability of a certain threshold value.
3. The web application can be integrated with local weather forecast center to generate early warning.
4. The window showing the probability of landslide is easy to understand.

1.5 Motivation

Despite being a frequent event, not getting enough attention; landslide has brought enough crisis and loss upon Rangamati in past few years. As the population grows and societies become more diverse, the economic and social damages caused by such events are likely to escalate unless sufficient consideration is paid at an early level, as growing anthropogenic activities in mountain areas will exacerbate communities' already established vulnerability.

As Machine Learning been solving a lot of critical problems lately we tried to predict the landslide of Rangamati using Machine Learning and generate early warning to minimize the loss of lives and assets. Saving lives and minimizing the loss of assets worked as our motivation of this work.

1.6 Contribution of the thesis

The areas of contribution of the thesis can be summarized as follows:

- Designing a web application to predict the landslide of a specific location in real time.
- Generated a map marking the location with probability of landslide more than 60%.

- Analyzed the rainfall data of 5 major landslides to see the impacts of rainfall in landslide.

1.7 Thesis Organization

The rest of the report is sequenced as follows:

- In Chapter 2 we discussed about the previous study conducted by the researchers and related works till date. We also discuss the technology used for data retrieving, different Machine Learning algorithms we applied during experiments, the web technologies we used to deploy etc.
- In Chapter 3 we described our proposed methods in details from the very first step of data collection to the very last step model deployment.
- In Chapter 4 we analyzed the dataset, visualized necessary graphs, analyzed the performance of the different Machine Learning algorithms on our dataset, select a model through analysis of the performance and also portrayed how the web application works fine.
- In Chapter 5 we concluded the work and also mentioned the future work that can be developed from this stage.

1.8 Conclusion

In this chapter, we tried to portrait an overall overview of our proposed system. The challenge we faced and the application of this work have been briefly discussed. The importance of the work is mentioned in the motivation section and contribution of the work is also stated.
In the next chapter, the works of the previous researchers in this area of research are explored.

Chapter 2

Literature Review

2.1 Introduction

In the previous chapter, we presented a summary and importance of the work. The contribution and application of the work were also discussed. A lot of researchers been working to predict landslide using Machine Learning and Artificial Intelligence for a while. In this chapter, we discussed important related researches of the researchers and outlined the structure we've worked on. Moreover, the key findings of the researches and the methods that are applied to extract the result are also discussed. Outcomes of the scholarly articles are analyzed to understand the current state.

2.2 Related Resaerch Works

A. Shirzadi et al. [2] proposed an Alternating Decision Tree (ADTree) based on the Bagging (BA), Multiboost (MB), Random Subspace (RS), and Rotation Forest (RF) ensemble algorithms for spatial prediction of shallow landslides in Bijar City, Kurdistan Province, Iran, using two scenarios of separate sample sizes and raster resolutions.

For Lishui City in Zhejiang Province, China, Y. Wang et al[3] used the Synthetic Minority Oversampling Technique (SMOTE) to increase the number of landslide samples for machine learning methods (i.e., Artificial Neural Network (ANN), Logistic Regression (LR), Support Vector Machine (SVM), and Random Forest (RF)) to generate high-quality landslide susceptibility maps.

To mitigate disastrous landslides, F. Ren and X. Wu [4] selected the Baishuihe landslide in the Three Gorges region as a case study in predicting displacement using the monitoring data and a Radial Basis Function-Support Vector Machine (RBF-SVM) model. Periodic precipitation, reservoir level, and groundwater level variability have had a significant impact on landslide displacement. Prediction results indicated that the RBF-SVM with the optimal parameters ς , ϵ and γ of 170, 0.05 and 0.04 can provide the best predictive accuracy, with the minimum and maximum absolute error values of 0.47 and 9.84 mm, respectively.

J. Tingyao and W. Dinglong[5] proposed a new Bayesian Network-based landslide stability estimation approach that used the K2 algorithm and Bayesian method to learn the configuration and parameters of the Bayesian network. A joint tree inference algorithm is used in the developed Bayesian network model to evaluate and quantify landslide stability under the effects of slope height, slope angle, bulk density, angle of internal friction, cohesion, and other factors.

Based on temporal and spatial sensor data, L. Xiao, Y. Zhang, and G. Peng [6] used data-driven algorithms to predict landslide susceptibility. Elevation, slope angle, slope aspect, plan curvature, vegetation index, built-up index, stream strength, lithology, precipitation rate, and accumulated precipitation index were among the ten landslide instability factors created. Support Vector Machines (SVM), Decision Tree (DT), Long Short Term Memory (LSTM) and Back Propagation Neural Network (BPNN) were implemented and their final prediction accuracies were compared. The experimental results showed that the prediction accuracies of SVM, DT, LSTM and BPNN in the test areas were 72.9%, 60.4%, 81.2% and 62.0% respectively.

K. Agrawal et al. [7] applied state-of-the-art techniques to correct the class imbalance in landslide datasets. They used different synthetic and oversampling techniques to a real-world landslide data collected from the Chandigarh–Manali highway to resolve the class-imbalance issue. Also, they applied several machine-learning algorithms to the landslide data set for predicting landslides and evaluating their algorithms. The area under the ROC curve (AUC) and the sensitivity index (d’) were used to evaluate different algorithms. Results suggested that Random Forest (RF) algorithm performed better compared to other classification techniques like Neural Networks (NN), Logistic Regression (LR), Support Vector Machines (SVM) and Decision Trees (DT).

2.3 Satellite Image

Photographs of Earth or other planets taken by artificial satellites are known as satellite imagery. Satellite images provide a good representation of what is going on around the world, especially over oceans where data gaps are common. They're basically the sky's eyes. Satellite images help to display what can't be seen or measured. Again, satellite images are taken for granted. There isn't much space for mistake. Satellite images have "first-hand" information that can be analyzed.[8]

2.3.1 MODIS/006/MCD12Q1 Dataset

The MCD12Q1 V6 product offers global ground cover forms resulting from six separate classification schemes at yearly intervals (2001-2016). It is calculated using MODIS Terra and Aqua reflectance data that have been supervised classified. The supervised classifications are then refined further with additional post-processing that incorporates prior knowledge and ancillary data.[9]

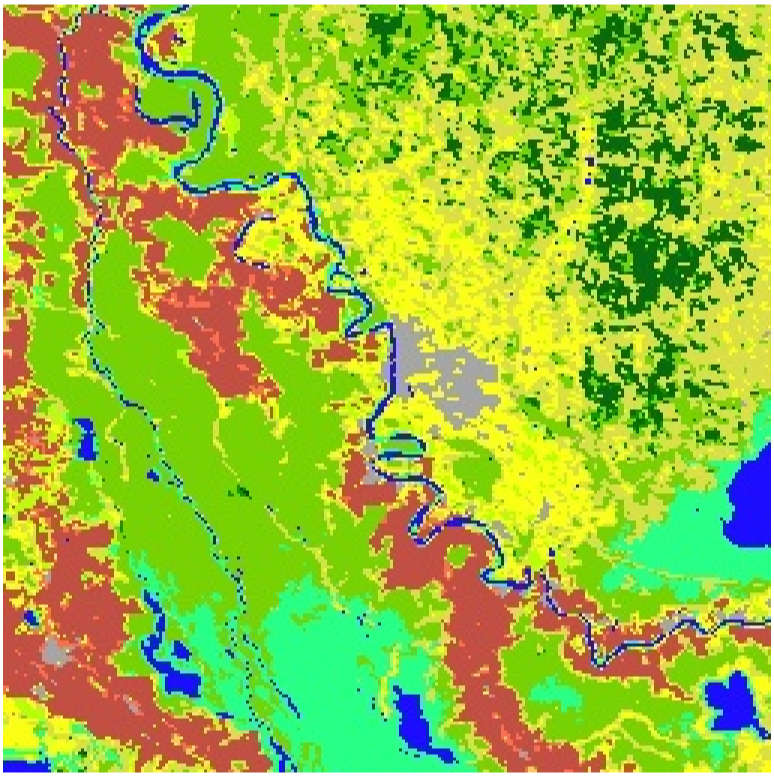

Figure 2.1: MODIS/006/MCD12Q1 Image

2.3.2 USGS/SRTMGL1_003 Dataset

The digital elevation data from the Shuttle Radar Topography Mission is the result of an international research project that produced digital elevation models on a near-global scale. Unlike other versions that include voids or have been void-filled with commercial sources, this dataset has been void-filled using open-source data (ASTER GDEM2, GMTED2010 and NED).[10]

Figure 2.2: USGS/SRTMGL1_003 Image

2.3.3 LANDSAT/LC08/C01/T1_TOA3 Dataset

Collection Landsat 8 1 calibrated top-of-atmosphere (TOA) reflectance from Tier 1. The picture metadata is used to extract the calibration coefficients.[11]

Figure 2.3: LANDSAT/LC08/C01/T1_TOA3 Image

2.3.4 MERIT/Hydro/v1_0_1 Dataset

MERIT Hydro is a modern global flow direction map derived from version 1.0.3 of the MERIT DEM elevation data and water body datasets with a 3 arc-second resolution (90 m at the equator) (G1WBM, GSWO and OpenStreetMap).

MERIT Hydro is the output of a new algorithm that almost instantly distinguishes river networks from noise caused by elevation data errors by extracting actual inland basins. In terms of flow aggregation area and river basin form, the built hydrography map indicates strong alignment with current quality-controlled river network datasets after a minimal amount of hand-editing. The positioning of river streamlines was realistically matched with current global river channel data obtained from satellites. 90% of GRDC gauges had a relative error in the drainage region of less than 0.05, supporting the consistency of the delineated global river networks. Flow accumulation region differences were mainly seen in arid river basins with depressions that are sometimes associated at high water levels, resulting in ambiguous watershed borders.

MERIT Hydro outperforms current global hydrography datasets in terms of geographic scope (between 90N and 60S) and small stream representation, thanks to the improved availability of high-quality baseline geospatial datasets..[12]

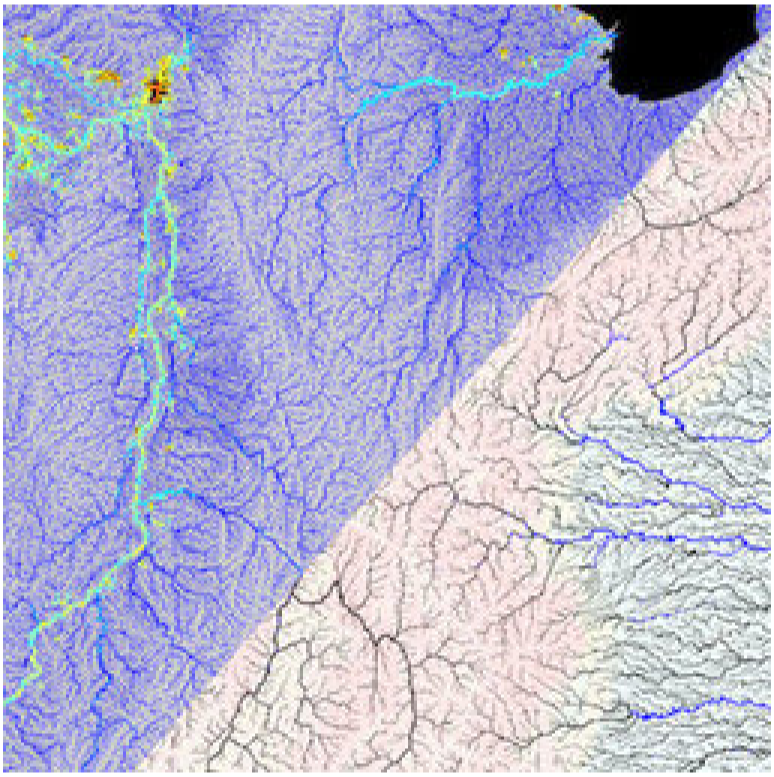

Figure 2.4: MERIT/Hydro/v1_0_1 Image

2.4 JAXA Climate Rainfall Watch

The need to track precipitation extremes from space is well understood, especially in areas where ground-based measurements are scarce or non-existent. In the Global Precipitation Measurement (GPM) mission, the Japan Aerospace Exploration Agency (JAXA) established the Global Satellite Mapping of Precipitation (GSMaP). The GSMaP Near-real-time Rainfall Product was provided by the JAXA as part of the World Meteorological Organization's (WMO) Space-based Weather and Climate Extremes Monitoring (SWCEM). Based on JAXA's experiences in the SWCEM, the JAXA's Earth Observation Research Center (EORC) has launched a website called "JAXA Climate Rainfall Watch," which uses GSMaP data to provide updates on severe heavy rainfall and drought around the world. By viewing cumulative rainfall on a temporal scale (daily, pentad, weekly, 10-days, and monthly), indices related to intense heavy rainfall (percentiles) and drought, one can easily track global extreme weather and environment.[13]

2.5 Google Earth Engine API

Google Earth Engine is a petabyte-scale collection of satellite imagery and geospatial information. 1st In reality, it's far more than a large-scale catalog; it gives users geospatial analytical capabilities as well as the ability to create their own custom pages and analytical resources that can be shared with others. It also allows researchers to share as well as submit their own personal data. Google's cloud-based architecture makes all of this possible.[14]
Google Earth Engine Application Programming Interface (API) is a cloud-based geospatial modeling tool that makes large-scale cloud computing capabilities accessible to the general public. Users will use this sophisticated geospatial computing tool to process both remotely sensed maps and vector datasets. Users will use this feature to look at societally relevant problems like drought, erosion, natural disasters, disease, food security, water management, climate monitoring, environmental monitoring, and more.[15]

2.6 Google Map API

The Google Maps Platform is a series of APIs and SDKs that developers can use to integrate Google Maps into mobile applications and web sites, as well as retrieve data from Google Maps. The Google Maps API is a brilliant piece of Google technology that allows one to embed the power of Google Maps directly on website. It allows one to customize the look and feel of the map to match the theme of site and incorporate related material that guests would find useful.

2.7 Machine Learning Algorithms

2.7.1 K-Nearest Neighbors

K-Nearest Neighbors is a basic but important classification algorithm in Machine Learning [16]. In the supervised learning domain, it sees applications like data processing, pattern recognition and intrusion detection, to name a few. It's widely used in real-world environments and it's non-parametric, which means it doesn't make any claims about how data is delivered. Prior data, also known as training data, is given, which categorizes coordinates according to an attribute. It selects n centroids at first, where n is the number of class labels to be predicted. Data points that are more closely related to one centroid are grouped together. Each data point has been allocated to one of the clusters. The Euclidean distance formula is used to measure the distance.

2.7.2 Decision Trees

The Decision Tree algorithm is one of the most widely used supervised machine learning algorithms for solving classification and regression problems. The decision tree addresses the problem by using the tree representation, where the leaf node corresponds to a class mark and attributes are represented on the internal node of the tree, and the goal is to build a model that forecasts the value of a target variable. The source set, which is the tree's root node, is divided into subsets, which are the tree's descendant children, to create a tree. The splitting is based on a collection of splitting rules based on classification features. Decision trees are one of the most common machine learning algorithms due to their comprehensibility and simplicity. [17]

2.7.3 Support Vector Machine

Support vector machines are supervised machine learning algorithms for classification and regression.[18] They're mostly used to solve classification issues. Each acquired data instance is plotted as a point on an n-dimensional space or graph by a support vector algorithm, where n is the total number of data features present.

The value of each data point is represented by an unique coordinate on the graph. To separate the data instances, the SVM constructs an n-1 dimensional hyperplane. The kernel trick is a method used in the algorithm that converts data into an optimal boundary that is used to differentiate the instances.

2.7.4 Linear Regression

By fitting a linear equation to observed data, linear regression attempts to predict the relationship between two variables. One variable is regarded as an independent variable, and the other is regarded as a dependent variable. A modeler can first decide whether or not there is a relationship between the variables of interest before attempting to fit a linear model to observed data. This does not actually mean that one variable affects the other, but rather that the two variables have a meaningful relationship. A scatterplot is a useful method for calculating the strength of a relationship between two variables. If the suggested predictive and dependent variables do not seem to be related, applying a linear regression model to the data is unlikely to yield a useful model. The correlation coefficient, which is a value between -1 and 1 representing the frequency of the relationship of the measured data for the two variables, is a useful numerical indicator of association between two variables. [19]

2.7.5 Random Forest

The supervised learning approach is used by Random Forest, a well-known machine learning algorithm [20]. In machine learning, it can be used for both classification and regression problems. It is based on ensemble learning, which is a method of combining multiple classifiers to solve a complex problem and improve the accuracy of the model. The algorithm produces several decision trees for the prediction task, thus the term forest. Rather than relying on a single decision tree, the random forest takes the projections from each tree and predicts the final performance based on the plurality of votes. The more trees in the landscape, the more precise it is, and the problem of overfitting is avoided.

2.7.6 Bayesian Regression

We formulate linear regression using probability distributions rather than point estimates from a Bayesian perspective. The answer is believed to be drawn from a probability distribution rather than being calculated as a single value.
The result is generated by a normal (Gaussian) distribution with a mean and variance. In linear regression, the mean is calculated by multiplying the weight matrix by the predictor matrix. The variance equals the standard deviation squared (multiplied by the Identity matrix because this is a multi-dimensional formulation of the model).
The aim of Bayesian Regression is to evaluate the posterior distribution for the model parameters, not to find a single "best" value for the model parameters. Not only is the answer based on a probability distribution, but the model parameters are often considered to be based on one. The train affects the posterior likelihood of the model parameters. [21]

2.7.7 AdaBoost

The AdaBoost algorithm, which stands for Adaptive Boosting, is a Boosting technique used as an Ensemble Method of Machine Learning. The weights are reassigned to each instance, with higher weights allocated to instances that were incorrectly labelled. This is called Adaptive Boosting. In supervised learning, boosting is used to reduce bias and variation. It is based on the concept of sequential growth of learners. Each successive learner, with the exception of the first, is grown from previously grown learners. In other words, weak students are transformed into good students. Since the Adaboost algorithm acts under the same principle as boosting, there is a minor variation in how it operates. [22]

2.7.8 Gradient Tree Boosting

Gradient boosting is a type of boosting used in machine learning. It is based on the assumption that when the best possible next model is paired with previous ones, the average prediction error is minimized. To mitigate error, the main concept is to set the goal outcomes for the next model. The target result for each case in the data is determined by how much adjusting the forecast for that case affects the average prediction error:

- If a marginal improvement in a case's estimation results in a significant reduction in error, the case's next target outcome is a high value. The error would be reduced if the current model's predictions come close to its targets.
- If a slight change in a case's forecast produces no change in error, the case's next target result is zero. Changes to this prediction have little impact on the mistake.

Gradient boosting gets its name from the fact that each case's target outcomes are determined by the gradient of the error with respect to the forecast. In the space of potential predictions for each training scenario, each new model takes a step in the direction that minimizes prediction error. [23]

2.7.9 Neural Network

A neural network is a collection of algorithms that tries to explain fundamental relationships in a set of data by mimicking how the human brain functions. Neural networks, in this context, refer to structures of neurons that may be endogenous or artificial in nature. Since neural networks can respond to evolving data, they can provide the best possible outcome without requiring the performance parameters to be redesigned. The artificial intelligence-based idea of neural networks is quickly gaining traction in the development of trading systems. [24]

2.8 Web Technology

2.8.1 Frontend Technology

HTML

HTML (HyperText Markup Language) is the most popular markup language for documents that will be displayed in a web browser. Online browsers translate

HTML documents received from a web server or locally saved files into multimedia web pages.HTML (HyperText Markup Language) is the most popular markup language for documents that will be displayed in a web browser. Online browsers translate HTML documents received from a web server or locally saved files into multimedia web pages. HTML was created to provide hints for document presentation and to determine the semantic structure of a web page. HTML elements are the components that make up HTML pages. HTML structures may be used to inject photos and other objects into the rendered tab, such as interactive types. HTML enables the creation of structured documents by denoting structural semantics for text such as headings, columns, links, quotes, and other artifacts. Browsers do not see the HTML tags, but they are used to read the text on the website.

CSS

CSS is a style sheet language for specifying how a text written in a markup language like HTML should look. CSS, like HTML and JavaScript, is an essential part of the World Wide Web. CSS is a style sheet that separates presentation from text, allowing you to change the interface, colors, and fonts. By defining the relevant CSS in a separate .css file, which eliminates redundancy and duplication in the structural material and enables the .css file to be cached to increase page load speed between the pages with same format and similar file, this separation helps improve content usability, provide more consistency and control in the specification of presentation characteristics, and encourage many web pages to share formatting by specifying the relevant CSS in a separate .css file, which helps improve content accessibility, provide more flexibility and control in the specification of presentation characteristics and enable several web pages.

2.8.2 Backend Technology

Flask

Flask is a microweb framework based on Python. Since it does not require the use of any special tools or repositories, it is referred to as a microframework.

It is missing a database abstraction layer, type validation, and other components that depend on third-party libraries to perform basic tasks. Extensions, on the other hand, can be used to integrate application features into Flask as if it were built-in. Extensions are available for object-relational mappers, form validation, upload management, transparent authentication technologies, and other framework-related tools. Flask is made up of the following elements:

1. **WSGI:** For building Python web applications, the Web Server Gateway Interface (WSGI) has become the industry standard. The Network Service Gateway Interface (WSGI) is a standard for creating a universal interface between a web server and a web application.

2. **Werkzeug:** It's a WSGI toolkit for handling requests, artifact responses, and other basic activities. This allows you to build a web application on top of it. Werkzeug is a key component of the Flask system.

3. **Jinja2:** Jinja2 is a popular Python templating engine. To make dynamic web pages, a web templating framework blends a blueprint with a specific data source.

2.9 Conclusion

Researchers' research into applying a machine learning algorithm to predict landslides is addressed in this chapter. The experimental techniques, data retrieval method, flaws and algorithms used for the results are described. Machine learning algorithms that have been proposed are also discussed. Traditional machine learning algorithms that are used in this context been shortly discussed. The methodology used in this study is briefly discussed in the following sections.

Chapter 3

Methodology

3.1 Introduction

Researchers have been conducting various forms of research for the past decade in order to effectively and reliably apply Artificial Intelligence to predict the landslide. These studies yielded significant results and had an effect on the current landslide prediction scheme. Machine Learning and Artificial Intelligence are being used to predict landslides. The results of previous researches were summarized in the previous chapter. The research methodology is examined, as well as the shortcomings that have been also identified. The methodology used to obtain the results in this work is briefly explained in this chapter.

3.2 Methodology Overview

Initially, historical data was collected. We preprocessed the data and divided it into two groups: training and testing. Historical data was analyzed, and visualizations depicted the roles of various features in landslide events. The preprocessed data that had been separated for training purposes was yet again used to train various machine learning models.

Using Google Map API the coordinates of a location was fetched and the fetched data was sent to Google Earth Engine to extract the value of the feature from different satellite images. Land Cover feature value was extracted from $MODIS/006/MCD12Q1$ Dataset. Elevation, Slope and Aspect feature values were collected from $USGS/SRTMGL1\_003$ Dataset. NDVI feature value was calculated from $LANDSAT/LC08/C01/T1\_TOA3$ Dataset. Flow Accumulation Area was collected from $MERIT/Hydro/v1\_0\_1$ Dataset, using which TWI and SPI were calculated using the Slope value that was extracted earlier. Rainfall data was collected from JAXA Global Rainfall Watch. NDVI, TWI and SPI were calculated using the formula given below:

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

$$SPI = A_s tan\beta$$

$$TWI = ln(\frac{\alpha}{tan\beta})$$

Here,

$A_s = Flow\ Accumulation\ Area\ (m^2)$

$\beta = The\ local\ slope\ gradient\ (radian)$

$\alpha = Cumulative\ upslope\ area\ draining\ through\ a\ point\ (m^2)$

Based on the performance of existing models in predicting with the current data, which was derived from various satellite images using Google Earth Engine, a model was chosen.

Frontend and backend technologies were used to create a web application. As Frontend we used HTML, CSS along with Google Map's API were used. As Backend Flask was used which is based on python. With the aim of having the best prediction performance, the chosen model was deployed to the web application.

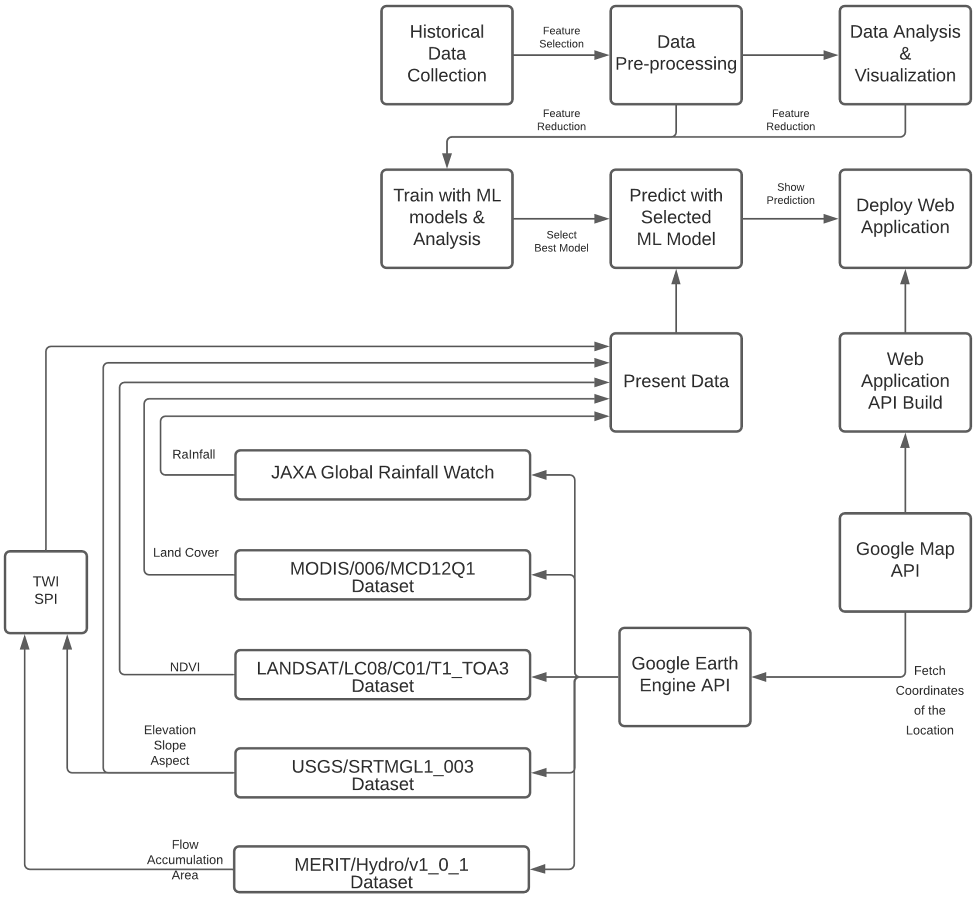

Figure 3.1: Overview of Methodology

3.3 Implemented Method

The implementation process has been defined in this section in a step-by-step manner, as seen in Figure 3.1.

3.3.1 Data Collection

Data collection is the single most important step in solving any machine learning problem. Many researchers and computer scientists, though, find it to be a huge roadblock. Data analysis, which consists mostly of data collection, data labeling, and upgrading existing data or models, usually takes a long time.

We faced huge problem collecting the dataset due to unavailability and insufficiency of data. Other than 4 major landslides date backs 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.

With lack of proper inventory of landslide we had to use third party dataset[2] published in 2020 based on the local inventory stated above.

3.3.2 Feature Selection

Feature selection is a technique for reducing the number of input variables in a predictive model. To decrease the computing cost of simulation and, in some cases, to improve the model's precision, the number of input variables may be decreased. The relationship between each input variable and the target variable is determined using statistics, and the input variables with the best relationship with the target variable are selected. This methods can be simple and effective since the choice of statistical metrics is dependent on the data type of both the input and output variables. The aim of feature selection is to reduce training time, escape the dimensionality curse, and simplify the machine learning model.

We considered 8 features 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 plotted a correlation matrix to examine the correlation of primary features in order to identify the features that are significant as well as insignificant features.

[1]https://doi.org/10.3390/data5010004
[2]https://github.com/yrabby/XGBoost_Paper/blob/master/Dis2.csv

The efficiency of a model can be improved by removing features that aren't necessary for prediction. However, no such features are present in our work, so no functionality is dropped or changed.

3.3.3 Data Preprocessing

The stage of any Machine Learning process where the data is translated, or encoded, to make it easier for the machine to process is known as data preprocessing. To put it another way, the algorithm can now easily comprehend the properties of the results. Raw data is incomprehensible to machines. Furthermore, processing string or text data does not allow for prediction. To classify, algorithms require data to be plotted in a graph or plane. As a consequence, the obtained raw data must be converted to numerical data before machine learning algorithms can be used to achieve the desired result. Data cleaning is an essential aspect of conventional data preprocessing measures. Many sections of the data may be useless or incomplete. In order to handle this section, data cleaning is performed. It involves working with data that is imperfect, noisy and so on. However, all of the data we collected were numerical, and there were no gaps in the data. As a result, data cleaning isn't needed in this case. We normalized our dataset to improve performance since the values were differing in many ranges. There were no decrease of dimensionality.

3.3.4 Data Analysis

Data analysis is an essential aspect of the data collection process. Data analysis aids data engineers in choosing the final features to be used in machine learning algorithms that predict the outcomes. The relation between features is clarified in this process. Furthermore, data analysis aids the data engineer in identifying the attributes or factors that can be removed. The result of the analysis aids in interpreting the story that the data is trying to tell and determining its relevance. In the following part, Chapter 4, the results of the data analysis using the obtained dataset are presented. In that section, the results of the data analysis are briefly discussed. However, the most important findings were that each of the features chosen had an impact on a portion of the final result. As a result, no features are removed or changed until the data is fed into the machine learning model.

3.3.5 Model Evaluation and Selection

Using powerful machine learning libraries like scikit-learn, it's easy to suit a range of machine learning models on a predictive modeling dataset. As a result, one of the most difficult aspects of applied machine learning is choosing which model to use for a particular problem from a wide range of possibilities. Model selection in machine learning is a method for comparing models of various types, as well as models of the same type with different model hyperparameters. The aim of model evaluation is to evaluate the model's generalization error. An effective machine learning model performs well not only on data that was seen during training, but also on data that was not seen during training. As a result, we should be fairly sure that a model's performance would not deteriorate when confronted with new data before releasing it into production.

Several machine learning models were trained with the collected preprocessed data for this work's model selection process. The following are some of the machine learning algorithms that have been trained and evaluated:

- 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)

Random Forest(RF) model is chosen and deployed using the Web Application after the performance of the machine learning algorithms have been evaluated.

3.3.6 Web Application Development

Frontend Development

The technique of writing HTML, CSS, and JavaScript code for a website or Web application such that a user can directly access and communicate with it is known as front-end web development, also known as client-side development. The challenge with frontend construction is that the approaches and techniques for building a website's frontend change all the time, necessitating the developer's constant understanding of how the field progresses. The aim of website design is to make sure that when people come to the site, they see content that is easy to read and understand. The challenge with frontend construction is that the approaches and techniques for building a website's frontend change all the time, necessitating the developer's constant understanding of how the field progresses. The aim of website design is to make sure that when people come to the site, they see content that is easy to read and understand. CSS beautifies the interface.

Backend Development

Backend architecture refers to the server side of development, which is mostly concerned with how the web works. The main responsibilities will include making changes and upgrades, as well as managing the site's functionality. This type of web development usually includes a server, an application, and a database. Backend developers write the code that allows the browser to interact with the database. A backend developer is in charge of things that aren't visible to the naked eye, such as databases and servers. Backend developers are also known as programmers or web developers in some cases.

The backend of the Web Application is developed using Python programming and the Flask microservice. Selected machine learning model that's been chosen based on performance is trained and saved as a pickle file. After that, the model is loaded into the backend codebase. In the backend, two APIs have been developed:

the main page API and the predict API. When anyone clicks on the map with the mouse, the information is retrieved via HTML form on the client-side of the Web Application and then fetched via API to the backend. When mouse is clicked coordinates of that location pass through the backend and extract the features from the satellite images and then passed through the trained model to get the prediction result. The result is then displayed on the application's frontend.

3.4 Conclusion

The system used to accomplish the work's goal is explained and explored in detail in this part. The process starts with data collection. A set of feature is selected to accomplish the procedure. Data analysis and visualization is done for better understanding of the features' importance and mutual correlations of the features. With the final selected features dataset is trained with different machine learning model. Based on the performance of the models, one model is selected to perform the prediction. A web application is developed and model is deployed in the web application for usability. The following chapter discuss the implementation results.

Chapter 4

Results and Discussions

4.1 Introduction

Result and Analysis is a section containing a synopsis of a study's main findings is developed, furthermore this section interprets and discusses the findings for readers. In the Outcome and Discussion section, the work's investigation and statistical analysis findings are presented. Based on the information obtained, it summarizes and makes recommendations. This section helps other researchers in deciding what is feasible and what can be predicted by using a particular methodology.

The method used to obtain the target result was thoroughly explored in the previous chapter. The data collection process, as well as the feature selection process and the interpretation of the collected data, are all clarified. Also discussed the evaluation of machine learning algorithms.

The outcome of the implemented method is depicted in this chapter. The core observations of the procedure, as well as the knowledge obtained from it, are also discussed. Figures are used to illustrate the data. The derived outcome is often accompanied by contextual analysis.

4.2 Dataset Description

The dataset consists of total 392 landslides events with equal amount of positive and negative sample. We chose 8 important features that are highly impacts landslide immensely, to predict the probability of landslide.

Figure 4.1 represents the distribution of 392 records according to two class. We can see that the distributions of data is balanced.

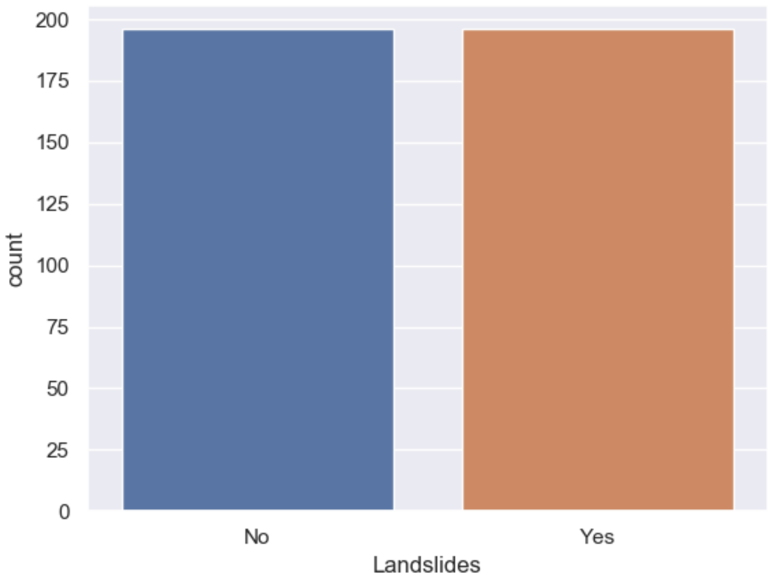

Figure 4.1: No. of Records

Figure 4.2 shows the correlation among features. It is clear that Slope, Elevation, NDVI, TWI and Rainfall are much closely related to prediction results than others. Despite being closely related to prediction results, NDVI of a specific location is almost constant for a long period. Again, TWI is calculated from Slope value, thus dependable. So now we can consider Slope, Elevation and Rainfall to be more impactful in event of landslide than other features.

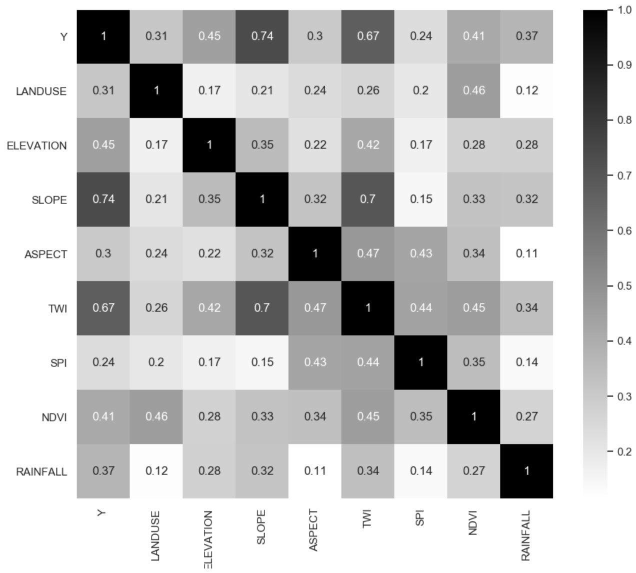

Figure 4.2: Correlation Matrix

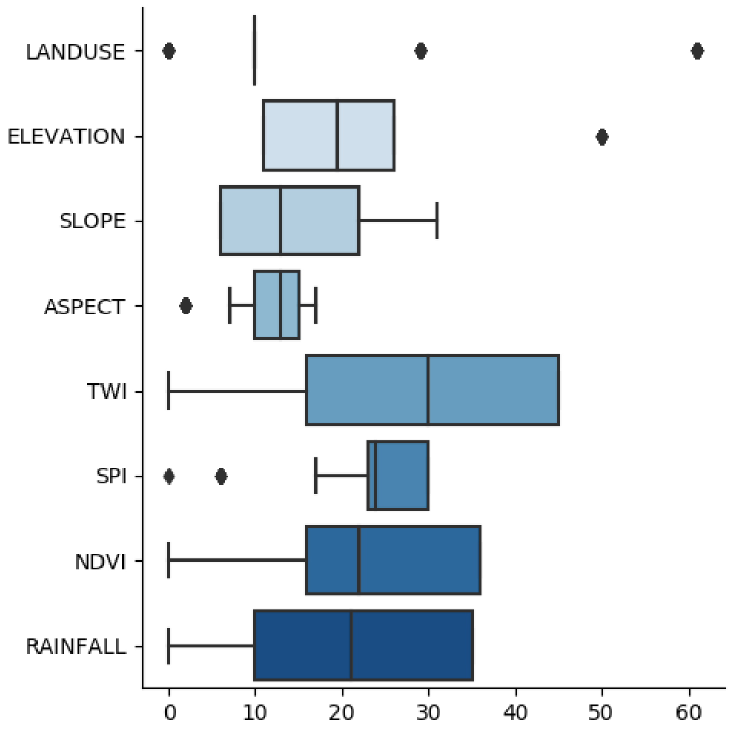

Figure 4.3: Features Data Distribution

Figure 4.3 illustrates the distribution of the features data. It can be seen that only a few features shows variety of range where many of them have a small range of data. This identifies that Rainfall plays an important role in the event of landslide.

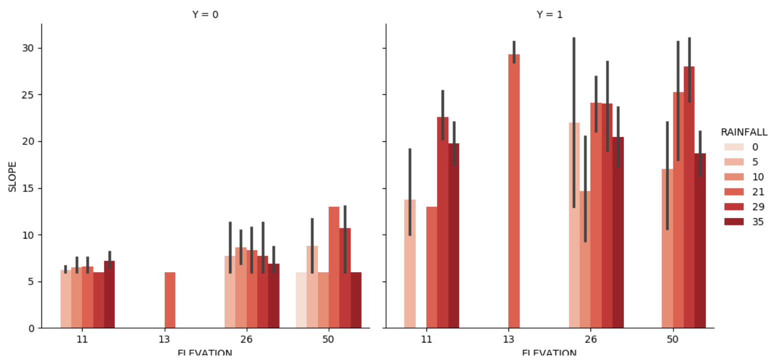

Figure 4.4: Impact of Rainfall, Slope and Elevation

Figure 4.4 shows us the impact of Rainfall, Slope and Elevation combined. It is clear from the graph that high Slope and heavy Rainfall can cause a landslide. High Elevation makes landslide more frequent along with high slope and heavy Rainfall.

4.3 Impact Analysis

Machine Learning has open a new horizon of incredibility. There's been a vast improvement on machine learning applications in daily life. The work we've done have many impacts in our daily life. Impact of the work are stated below:

1. With the help of web application one can easily know the landslide probability of a certain location.
2. The Web Application can generate early warning to avoid loss of lives and assets.
3. It can map the areas with high risk of landslide given a certain threshold value.

4.4 Impact of Rainfall in Landslide of Rangamati

Here we're mentioning 5 major landslides that took place in Rangamati. Source says, among these landslides 4 of them were caused by heavy rainfall and one happened due to digging. We collected rainfall data of each month date backs to one month and analyzed the result. We found something interesting. Our analysis match as per source says.

4 of the landslides caused by heavy rainfall as per source, we found there's been excessive rainfall of more than 200 mm for at least last 15 days before each landslide. The one landslide that happened due to digging as per source says recorded around 40 mm of rainfall last 15 days before the landslide.

Table 4.1: Major Landslides list

Date	Location		Cause	Rainfall Recorded Last 15 days
	Lattitude	Longtitude		
11/06/2007	22.64	92.15	Heavy Rainfall	221.91 mm
16/03/2017	22.65	92.15	Digging	39.86 mm
12/06/2017	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

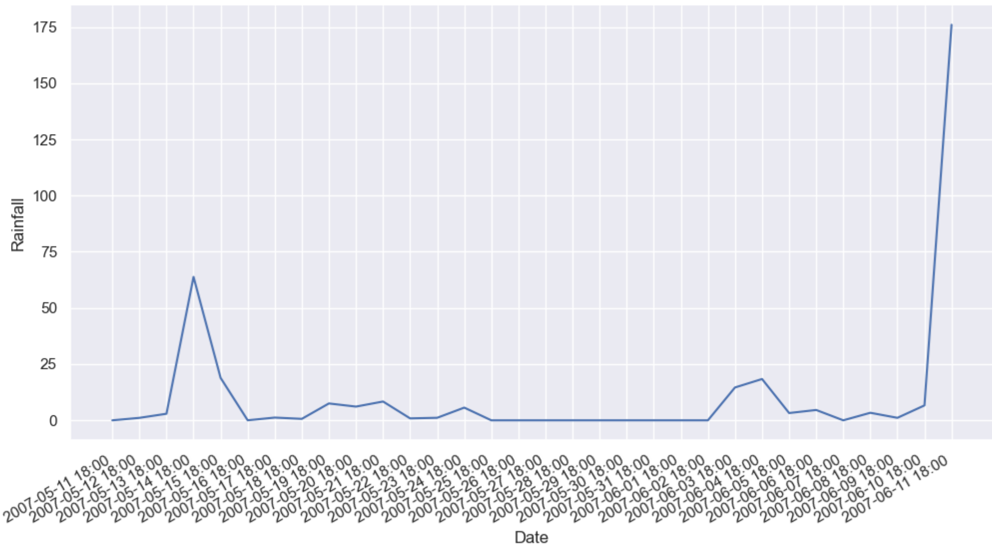

Figure 4.5: Rainfall Record(Day to Day)
11/06/2007
Latitude: 22.64 Longitude: 92.15

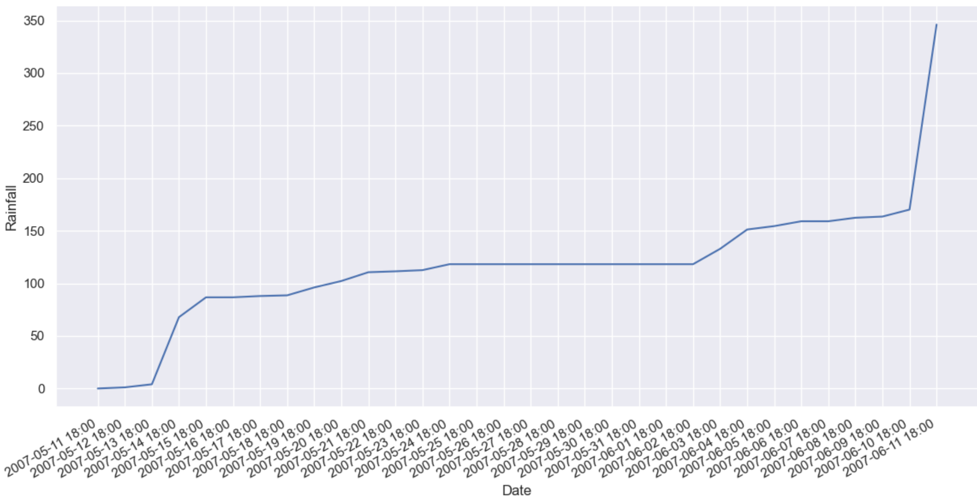

Figure 4.6: Rainfall Record(Cumulative)
11/06/2007
Latitude: 22.64 Longitude: 92.15

In Figure 4.5 & Figure 4.6 shows Rainfall records of location 22.64N,92.15E date backs to 11/05/2007-11/06/2007 has been shown in graphs. A total amount of 221.91 mm rainfall was recorded in the last 15 days. At least 128, including at least 59 children, were dead, with more than 150 people hospitalized in this landslide. [25] In this landslide rainfall played an important role.

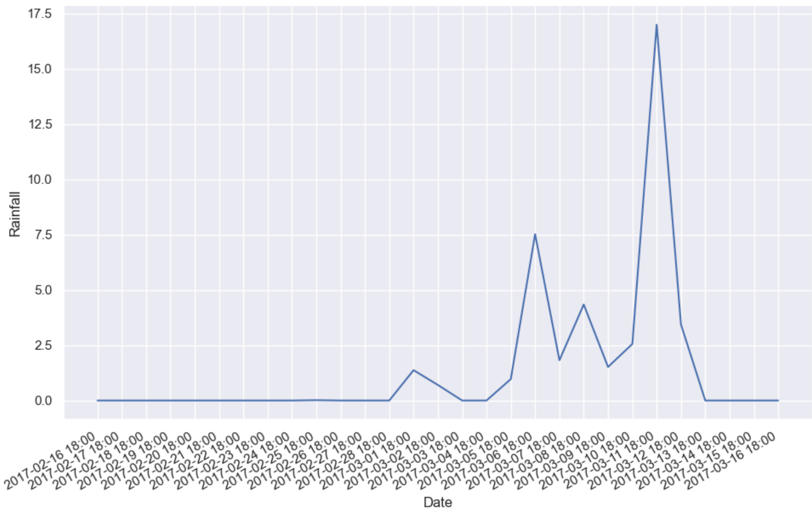

Figure 4.7: Rainfall Record(Day to Day)
16/03/2017
Latitude: 22.65 Longitude: 92.15

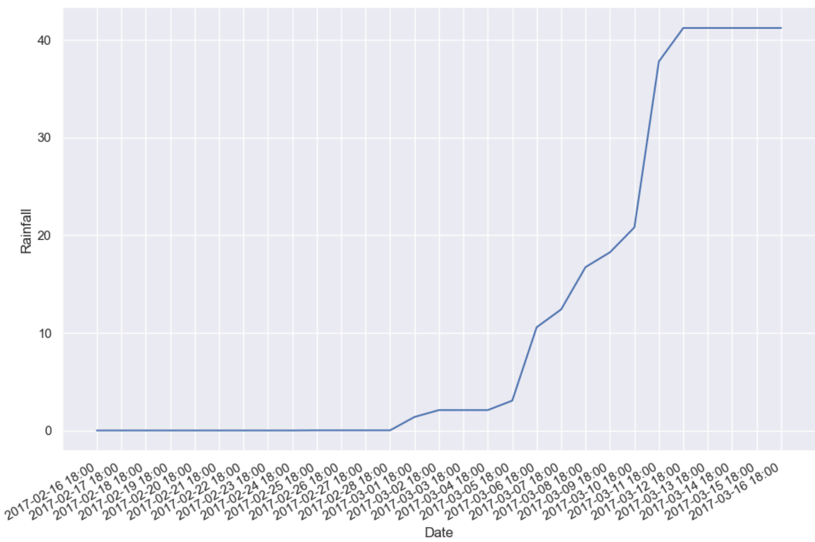

Figure 4.8: Rainfall Record(Cumulative)
16/03/2017
Latitude: 22.65 Longitude: 92.15

In Figure 4.7 & Figure 4.8 shows Rainfall records of location 22.65N,92.15E date backs to 16/02/2007-16/03/2007 has been shown in graphs. A total amount of 39.86 mm rainfall was recorded in the last 15 days. At least three people were killed in a landslide while building a wall on the foot of a hill.[26] This landslide took place due to digging. Rainfall didn't play any vital role here.

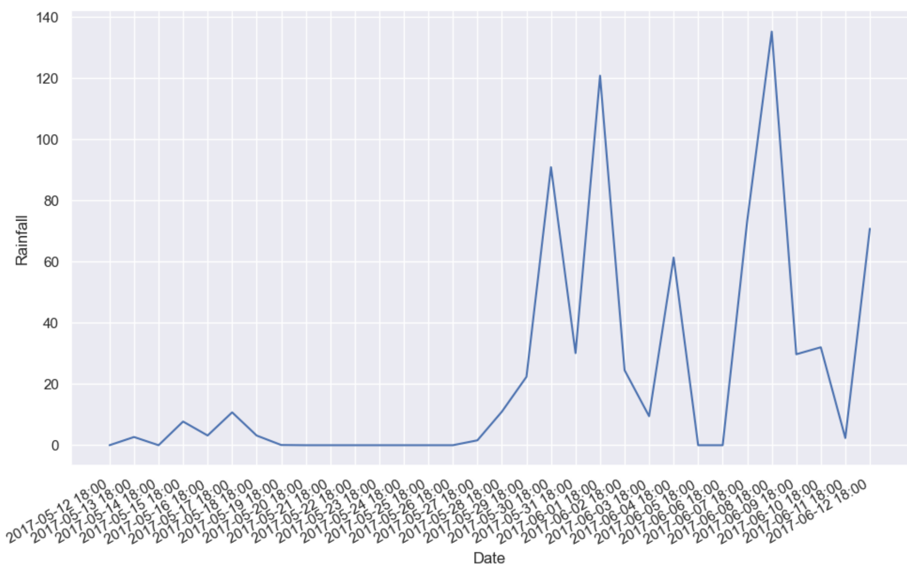

Figure 4.9: Rainfall Record(Day to Day)
12/06/2017
Latitude: 22.59 Longitude: 92.16

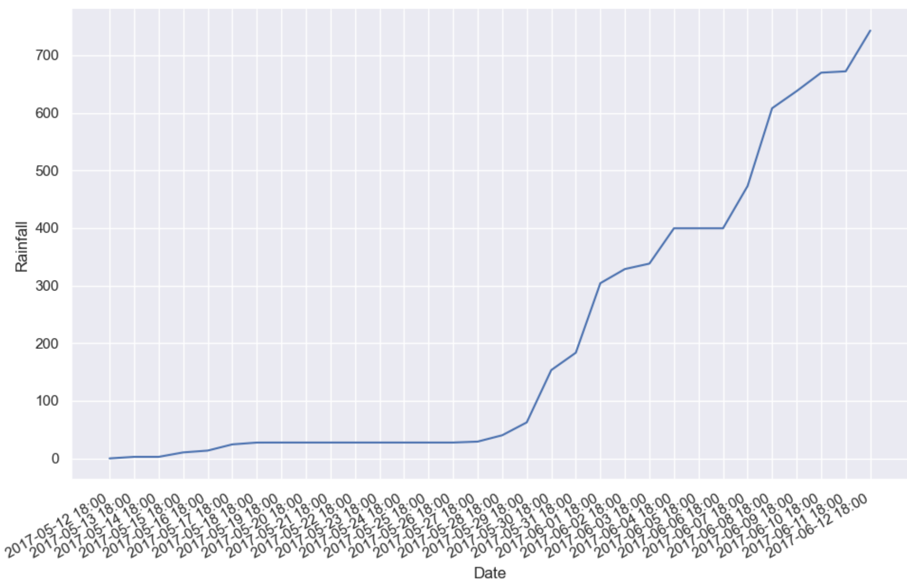

Figure 4.10: Rainfall Record(Cumulative)
12/06/2017
Latitude: 22.59 Longitude: 92.16

In Figure 4.9 & Figure 4.10 shows Rainfall records of location 22.59N,92.16E date backs to 12/05/2017-12/06/2017 has been shown in graphs. A total amount of 702.83 mm rainfall was recorded in the last 15 days. At least 100 people, including four army men, were killed in separate incidents of landslide in different upazilas of Rangamati.[27] In this landslide rainfall played an important role.

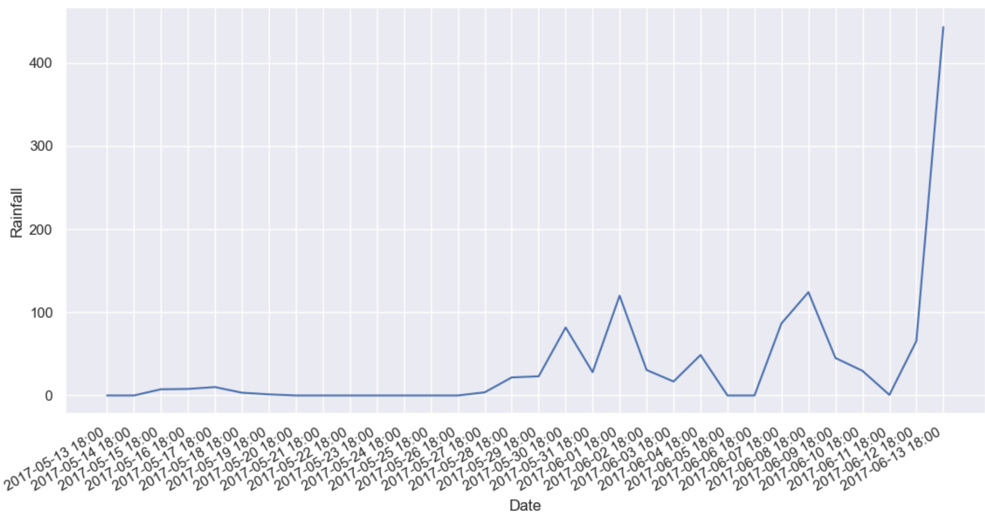

Figure 4.11: Rainfall Record(Day to Day)
13/06/2017
Latitude: 22.65 Longitude: 92.16

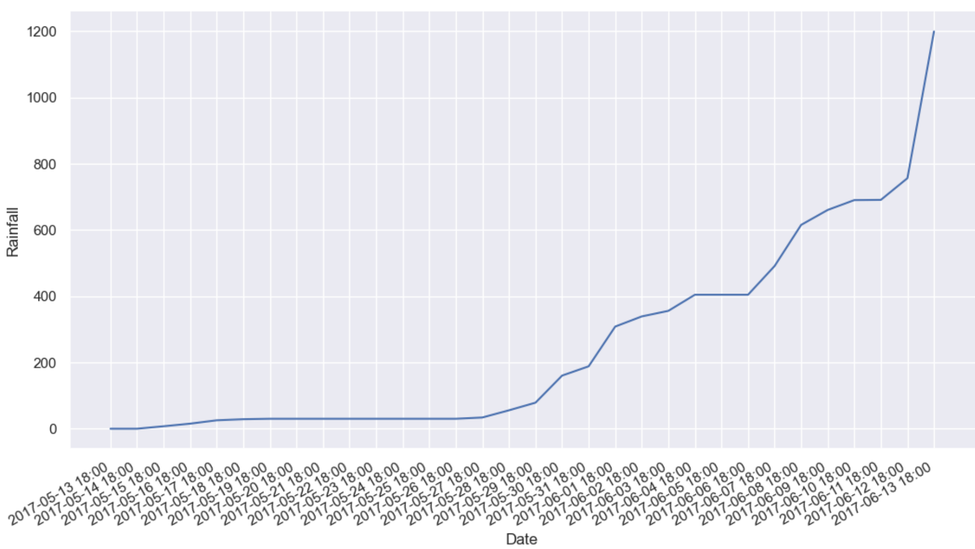

Figure 4.12: Rainfall Record(Cumulative)
13/06/2017
Latitude: 22.65 Longitude: 92.16

In Figure 4.11 & Figure 4.12 shows Rainfall records of location 22.65N,92.16E date backs to 13/05/2017-13/06/2017 has been shown in graphs. At least 100 people, including four army men, were killed in separate incidents of landslide in different upazilas of Rangamati.[27] In this landslide rainfall played an important role.

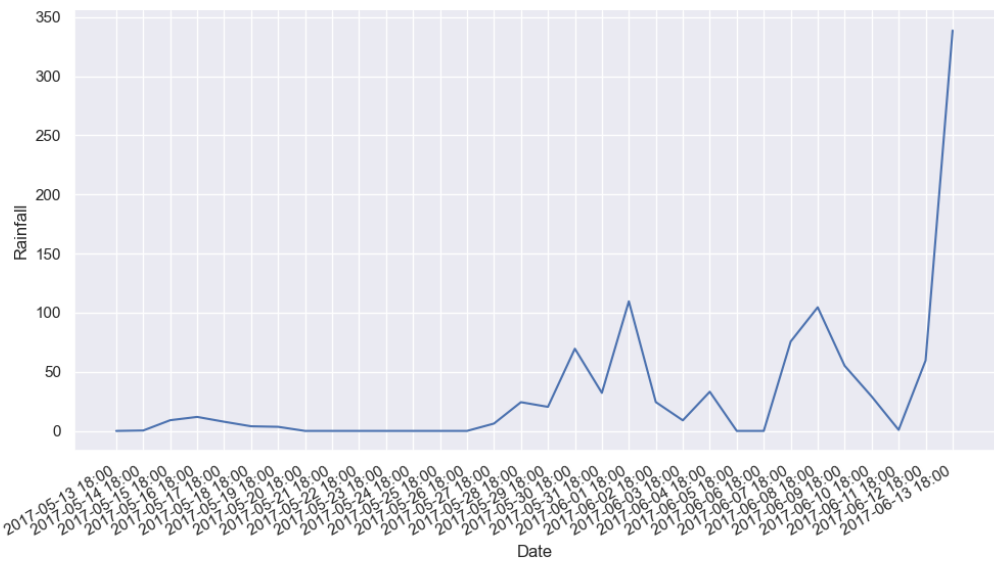

Figure 4.13: Rainfall Record(Day to Day)
13/06/2017
Latitude: 22.75 Longitude: 92.22

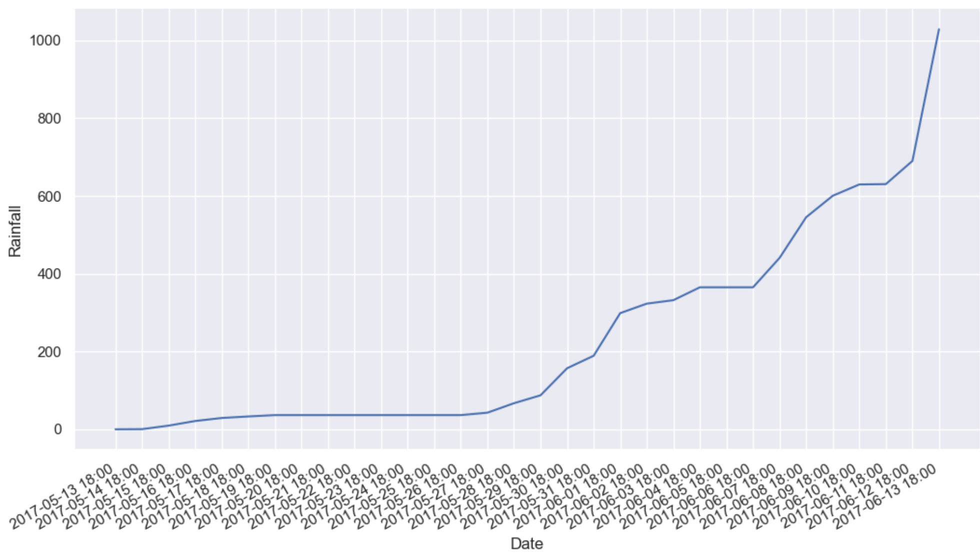

Figure 4.14: Rainfall Record(Cumulative)
13/06/2017
Latitude: 22.75 Longitude: 92.22

In Figure 4.13 & Figure 4.14 shows Rainfall records of location 22.75N,92.22E date backs to 13/05/2017-13/06/2017 has been shown in graphs. A total amount of 941.14 mm rainfall was recorded in the last 15 days. At least 100 people, including four army men, were killed in separate incidents of landslide in different upazilas of Rangamati.[27] In this landslide rainfall played an important role.

4.5 Evaluation of Performance

The evaluation of results is a crucial step in the machine learning process. However, it is a challenging challenge. As a consequence, it is essential to proceed with caution when using machine learning. A machine learning algorithm assessment is needed for any analysis. A machine learning algorithm calculated using an accuracy score metric may produce adequate results when compared to other metrics such as logarithmic loss or any other metric, but it may produce inferior results when compared to other metrics such as logarithmic loss or any other metric. Classification accuracy is often used to evaluate a learning model's performance; however, this is insufficient to completely evaluate a model.

We used the nine most common machine learning classifiers and trained them to test the performance of machine learning models against our collected and preprocessed dataset in our research. We measured the model accuracy, precision, recall, and F1 score of each trained model for the assessment process and to assess the best-performing model. We divided the dataset into 75% and 25% for training and testing, respectively, for high-quality performance measurement. Each of the models was trained on the shuffled 75% of data instances before being checked on the remaining 25%.

Table 4.2: 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

We can analyze from 4.2 that Random Forest(RF) provides the best performance with an accuracy of 0.88 along with the best precision, recall and F1 score.

4.6 Web Application

The selected model is implemented as a Web Application for better usability. Flask microservices are used to build the web application.

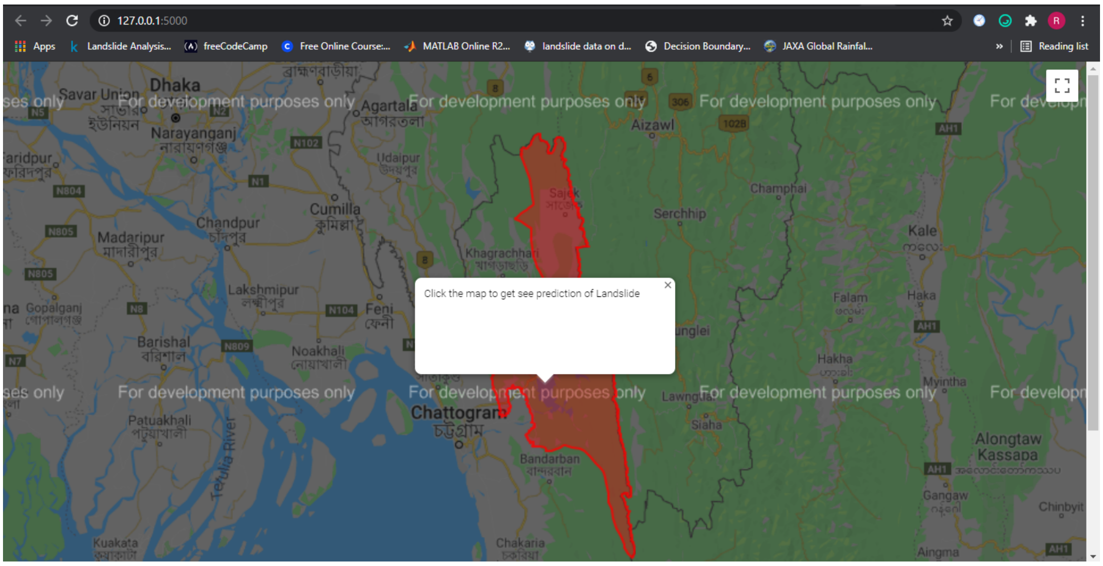

Figure 4.15: Home Page

Figure 4.15 displays the home page. It asks user to click on location to see the landslide probability.

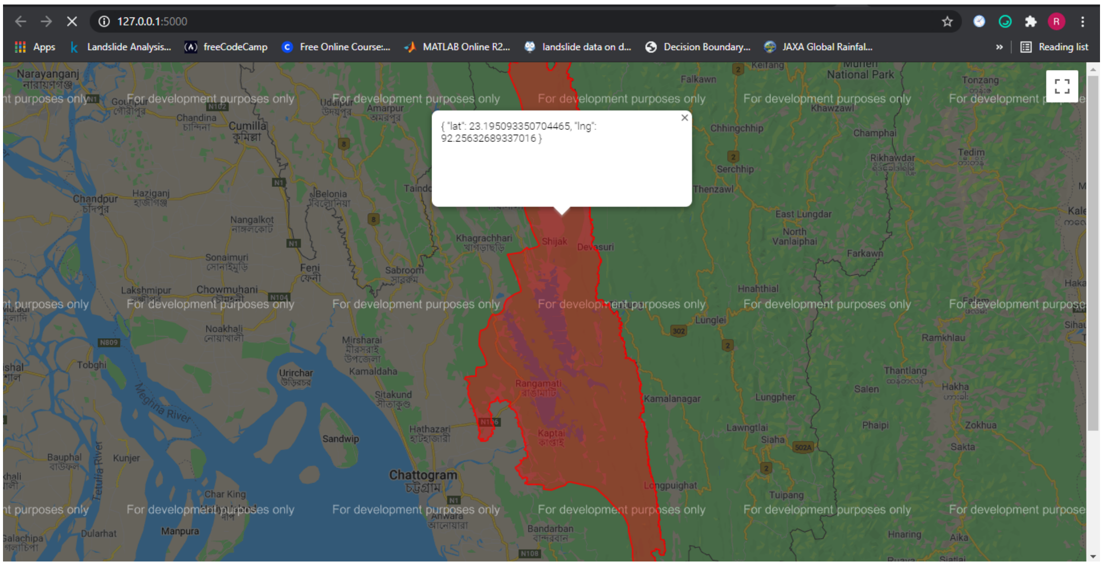

Figure 4.16: Loading Screen

Figure 4.16 shows the loading screen when mouse is clicked on the map. Clicking on the map the coordinate of that location passes through the Google Earth Engine to extract feature values.

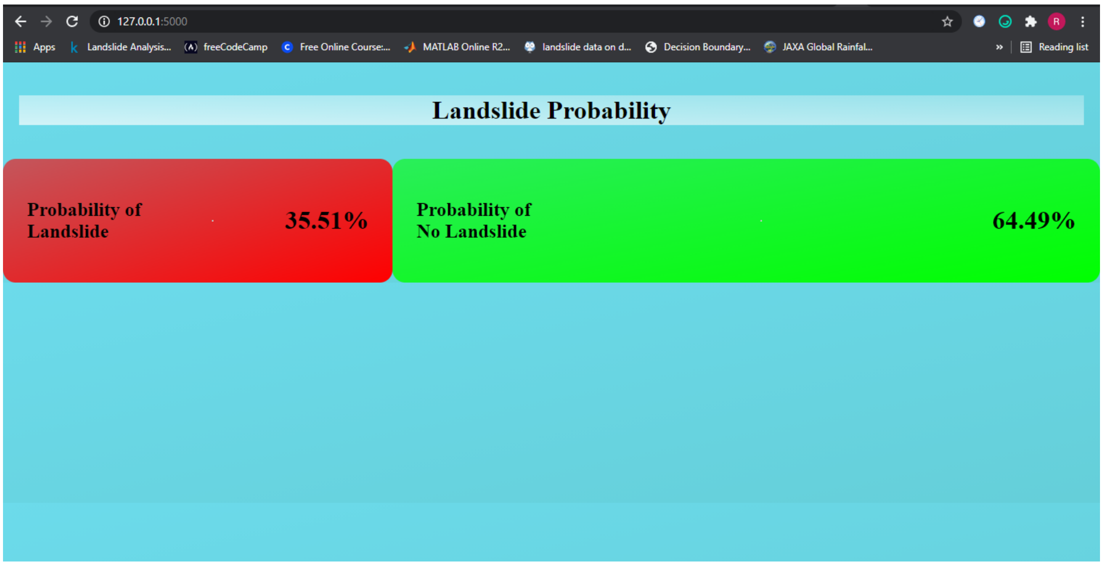

Figure 4.17: Displaying Prediction Result

Figure 4.17 displays the prediction result of the landslide. It displays landslide probability on a scale base. When there is more probability of having landslide the red portion of the scale gets bigger depending on the landslide probability.

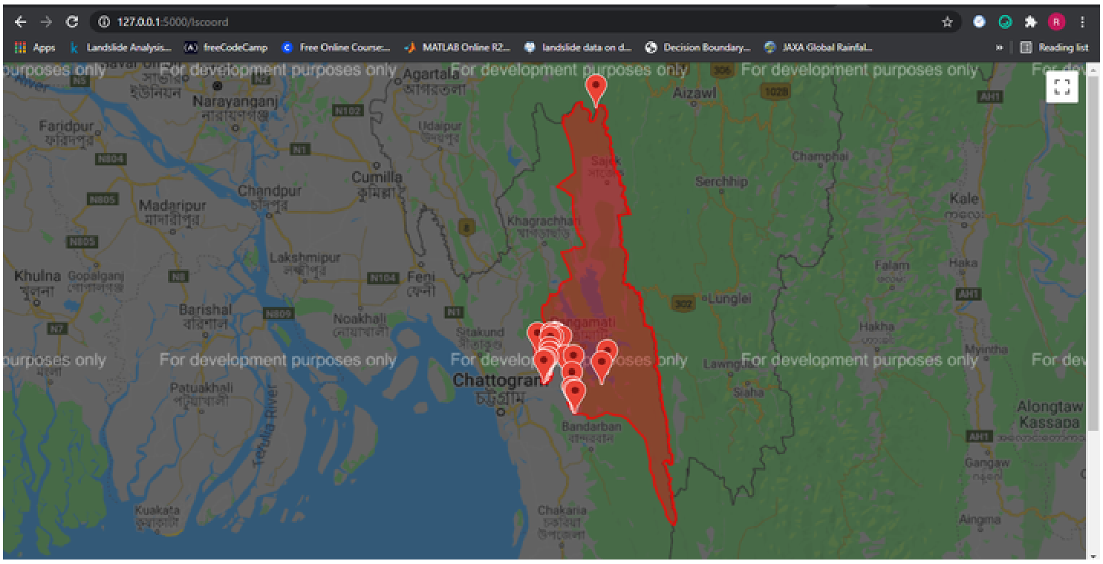

Figure 4.18: Location of landslide probability more than 60%

Figure 4.18 marks the location of the places with a landslide probability of more than or equal to 60%.

4.7 Conclusion

In principle, a model's predicted results will tell us how well it performs on unknown data. The most important thing we try to address is make predictions based on future results. Since each machine learning algorithm tries to solve a different problem with a different target using a different dataset, it's important to think about the context before choosing a metric. In this chapter, we've discussed the final result of our approach. The dataset is explained at the beginning of the chapter. In addition, the results of the data collection and main conclusions from the data are discussed. Following that, we present the model validation outcome as well as our findings from the research. A short comparison of our work with the work of previous researchers is also given. Figure of the interface is also presented which is intended to improve usability. We'll wrap up our research in the next chapter with a summary.

Chapter 5

Conclusion

5.1 Conclusion

Several landslides have occurred in Bangladesh's hilly districts in recent years. It demonstrates that a major landslide threat exists for such urban centers, and that many populations in Bangladesh are vulnerable to this hazard, which could cause serious destruction and socioeconomic effects in the future in many other regions. The urban centers that have been affected by landslides in the recent past are rapidly expanding and have a significant impact on the country's economic growth. It is therefore critical to create a model for landslide risk management and to provide a realistic understanding of the nature, severity and consequences of potential future landslide risks and casualties on susceptible populations living near landslide risk prone areas.

The approach to potential progress is to analyze past events and take appropriate proactive decisions based on what has been experienced. Our approach to predict landslide using machine learning is to predict the probability of having a landslide in real time and generate early warning.

The thesis was given a high-level description in the first chapter. In this chapter, the reader was exposed to related topics. The challenge faced and how this work was implemented are also quickly discussed. The significance of the piece, as well as its contribution, are explored in the inspiration section.

The researchers' research into using machine learning in the healthcare sector for disease detection is discussed in the following sections. The findings are described along with the experimental methods, data collection method, defects, and algorithms used in the literature. Algorithms for machine learning that have been proposed are also discussed. The use of traditional machine learning algorithms in this case was addressed briefly later.

The approach used to accomplish the work's goal is well explained and debated in chapter 3. From data collection to model deployment each and every step was explained in details.

We explained how our methodology generated the results we in Chapter 4. At the start of the chapter, the dataset is described. The findings of the data analysis as well as the key points drawn from the data are also discussed. Furthermore, we analyzed the impact of Rainfall causing these landslide. Following that, we presented the model validation results as well as the findings from the research.

A brief summary of our work is also given, along with previous researchers' work. The GUI was also depicted, with the aim of improving usability.

To predict the landslide of any certain place would require the historical landslide dataset of that particular region as landslides in a particular region most likely to be controlled by different factors differs from place to place. Moreover, historical dataset of one place won't be able to predict landslide of other places, as many geospatial factors are mainly reasonable for the occurance of landslides.

5.2 Future Work

There are several plans that can be pursued in the future. They are summarized below:

- Creating a proper inventory of all landslides so that the dataset will be more reliable.
- Improving the interface for more user friendly environment.
- Improving the site so that it can be integrated with the local forecast center and generate early warning.
- Working all over the country other than Rangamati.

References

[1] O. Korup and A. Stolle, ‘Landslide prediction from machine learning,’ *Geology today*, vol. 30, no. 1, pp. 26–33, 2014 (cit. on p. 1).

[2] A. Shirzadi, K. Soliamani, M. Habibnejhad, A. Kavian, K. Chapi, H. Shahabi, W. Chen, K. Khosravi, B. Thai Pham, B. Pradhan *et al.*, ‘Novel gis based machine learning algorithms for shallow landslide susceptibility mapping,’ *Sensors*, vol. 18, no. 11, p. 3777, 2018 (cit. on pp. 1, 6).

[3] 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 (cit. on p. 6).

[4] 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 (cit. on p. 6).

[5] 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 (cit. on p. 7).

[6] 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 (cit. on p. 7).

[7] K. Agrawal, Y. Baweja, D. Dwivedi, R. Saha, P. Prasad, S. Agrawal, S. Kapoor, P. Chaturvedi, N. Mali, V. U. Kala *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 (cit. on p. 7).

[8] *Satellite images.* [Online]. Available: `https://cimss.ssec.wisc.edu/oakfield/sat.htm` (cit. on p. 8).

[9] *Mcd12q1.006 modis land cover type yearly global 500m.* [Online]. Available: `https://developers.google.com/earth-engine/datasets/catalog/MODIS_006_MCD12Q1#description` (cit. on p. 8).

[10] *Nasa srtm digital elevation 30m nbsp;/nbsp; earth engine data catalog.* [Online]. Available: `https://developers.google.com/earth-engine/datasets/catalog/USGS_SRTMGL1_003` (cit. on p. 9).

[11] *Usgs landsat 8 collection 1 tier 1 toa reflectance*. [Online]. Available: `https://developers.google.com/earth-engine/datasets/catalog/LANDSAT_LC08_C01_T1_TOA` (cit. on p. 9).

[12] *Merit hydro: Global hydrography datasets nbsp;|nbsp; earth engine data catalog*. [Online]. Available: `https://developers.google.com/earth-engine/datasets/catalog/MERIT_Hydro_v1_0_1` (cit. on p. 11).

[13] [Online]. Available: `https://www.space4water.org/s4w/web/softwaretoolweb-app/jaxa-climate-rainfall-watch` (cit. on p. 12).

[14] M. Altaweel, *A look at google earth engine*, Mar. 2021. [Online]. Available: `https://www.gislounge.com/a-look-at-google-earth-engine/` (cit. on p. 12).

[15] *Earth engine api*, May 2021. [Online]. Available: `https://serc.carleton.edu/sp/library/google_earth/earth_engine_api.html` (cit. on p. 12).

[16] E. Fix, *Discriminatory analysis: nonparametric discrimination, consistency properties*. USAF school of Aviation Medicine, 1985, vol. 1 (cit. on p. 13).

[17] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip *et al.*, 'Top 10 algorithms in data mining,' *Knowledge and information systems*, vol. 14, no. 1, pp. 1–37, 2008 (cit. on p. 13).

[18] C. Cortes and V. Vapnik, 'Support-vector networks,' *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995 (cit. on p. 13).

[19] [Online]. Available: `http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm#:~:text=A%5C%20linear%5C%20regression%5C%20line%5C%20has,y%5C%20when%5C%20x%5C%20%5C%3D%5C%200).` (cit. on p. 14).

[20] T. K. Ho, 'Random decision forests,' in *Proceedings of 3rd international conference on document analysis and recognition*, IEEE, vol. 1, 1995, pp. 278–282 (cit. on p. 14).

[21] W. Koehrsen, *Introduction to bayesian linear regression*, Apr. 2018. [Online]. Available: `https://towardsdatascience.com/introduction-to-bayesian-linear-regression-e66e60791ea7` (cit. on p. 15).

[22] G. L. T. -, *Adaboost algorithm: Boosting algorithm in machine learning*, Mar. 2021. [Online]. Available: `https://www.mygreatlearning.com/blog/adaboost-algorithm/#:~:text=AdaBoost%5C%20algorithm%5C%2C%5C%20short%5C%20for%5C%20Adaptive,weights%5C%20to%5C%20incorrectly%5C%20classified%5C%20instances.` (cit. on p. 15).

[23] *Gradient boosting explained - the coolest kid on the machine learning block*, Dec. 2020. [Online]. Available: `https://www.displayr.com/gradient-boosting-the-coolest-kid-on-the-machine-learning-block/#:~:text=Gradient%5C%20boosting%5C%20is%5C%20a%5C%20type,minimizes%5C%20the%5C%20overall%5C%20prediction%5C%20error.&`

text=reduce%5C%20the%5C%20error.-,If%5C%20a%5C%20small%5C%20change%5C%20in%5C%20the%5C%20prediction%5C%20for%5C%20a%5C%20case,of%5C%20the%5C%20case%5C%20is%5C%20zero. (cit. on p. 16).

[24] J. Chen, *Neural network definition*, Dec. 2020. [Online]. Available: https://www.investopedia.com/terms/n/neuralnetwork.asp (cit. on p. 16).

[25] *2007 chittagong mudslides*, May 2021. [Online]. Available: https://en.wikipedia.org/wiki/2007_Chittagong_mudslides (cit. on p. 31).

[26] *3 killed in rangamati landslide*, Mar. 2017. [Online]. Available: https://www.dhakatribune.com/bangladesh/nation/2017/03/16/3-killed-rangamati-landslide (cit. on p. 32).

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