AVALANCHE FORECASTING USING MACHINE LEARNING

PROJECT REPORT

Submitted By:

Manju Rekha S
Jayalakshmi S
Indrajothi S
Ramya L S

CONTENTS

ABSTRACT

Conventional avalanche forecasting is practiced as a mix of deterministics treatment for snow and weather parameters and indictive logic to reach actual forecast decisions. Inductive logic of the scientific method dominates,making frequent use of iteration and redundancy to minimize decision uncertanities.The model gets the data from the IOT based sensors.After that we want to process those data using suitable Machine learning algorithms, then our model display whether the avalanche occur or not and how strenth it was.The project prevent the people from the avalanche by priory informing them there is a chance to the occurrence of avalanche or not.

1.INTRODUCTION:

The word Avalanche refers to snow and ice.It means a mass of snow,ice, rocks, slush falling rapidly down a mountain. Snow avalanches are amoung the most destructive natural hazards threatening human life, ecosystem,built structure,and landscapes in mountainous regions. Each year avalanche kills more than 150 people worldwide. The most common cause of death by avalanche is asphyxiation. If the person buried under an avalanche more than 15 minutes then there is no chance of survive. So, the life of the people in that region is difficult to live.

1.1 OVERVIEW:

To analyse the data, coming from different sensors we are applying various machine learning algorithms. If there is a chance of avalanche then the notification will be sent to people so that they can take decisions accordingly. Avalanche is leading cause of major event occur in winter or spring.This may cause severe disaster to human and many lives.We demonstrate of the use of Machine Learning to build the model which predicts the avalanche level.

1.2 PURPOSE:

The main objective is to prevent the people from the avalanche by priory informing them there is a chance to the occurrence of avalanche or not. The model gets the data from the IOT based sensors. After that we want to process those data using a suitable algorithm, then our model display whether the avalanche occur or not and how strength it was. To analyse the data coming from different sensors we are applying various machine learning algorithms. If there is a chance of avalanche then the notification will be sent to people so that they can take decisions accordingly.

2. LITERATURE SURVEY:

2.1 EXISTING PROBLEM:

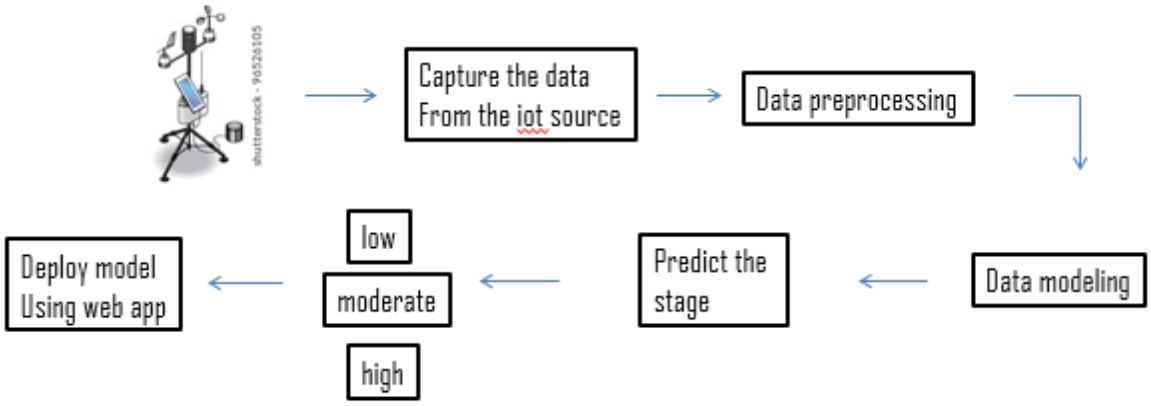
With the collected data of slope,forest density,snow density, Air temperature and wind, the main aim of the project is there is a chance of avalanche then the notification will be sent to people so that they can take decisions accordingly. Avalanche Warning Service uses avalanche problems in their bulletins to provide a compact, yet exact, description for the end user of the current conditions.The type of expected avalanche, the cause of the avalanche, the likelihood, expected size, amount of extra load that is necessary for triggering and finally where in the snowpack we expect the avalanche to be released

2.2 PROPOSED SOLUTION:

The model gets the data from the IOT based sensors. with the collected data from the model is build with the machine learning algorithm and proposed steps applied to those data, then our model display whether the avalanche occur or not and how strength it was. If there is a chance of avalanche then the notification will be sent to people so that they can take decisions accordingly. Here with these steps we proceed our model, first the data is collected with the specific resources. Secondly the collected data is preprocessed by importing libraries, importing dataset additional structuring data using data visualization, thirdly our model is build with the train and test technique and evaluated whether our model prediction is correct or not. Then our model is deployed , The web application created using html code and python code.

3. THEORETICAL ANALYSIS:

3.1 BLOCK DIAGRAM



3.2 HARDWARE/SOFTWARE DESIGNING:

For hardware we need IOT based sensors. With the IOT based sensors We collect the data which is given as input to the web page
For software we need compatible operating system for python, HTML and CSS, software used for python implementation and web deployment

1) Numpy, pandas, matplotlib
2)Flask

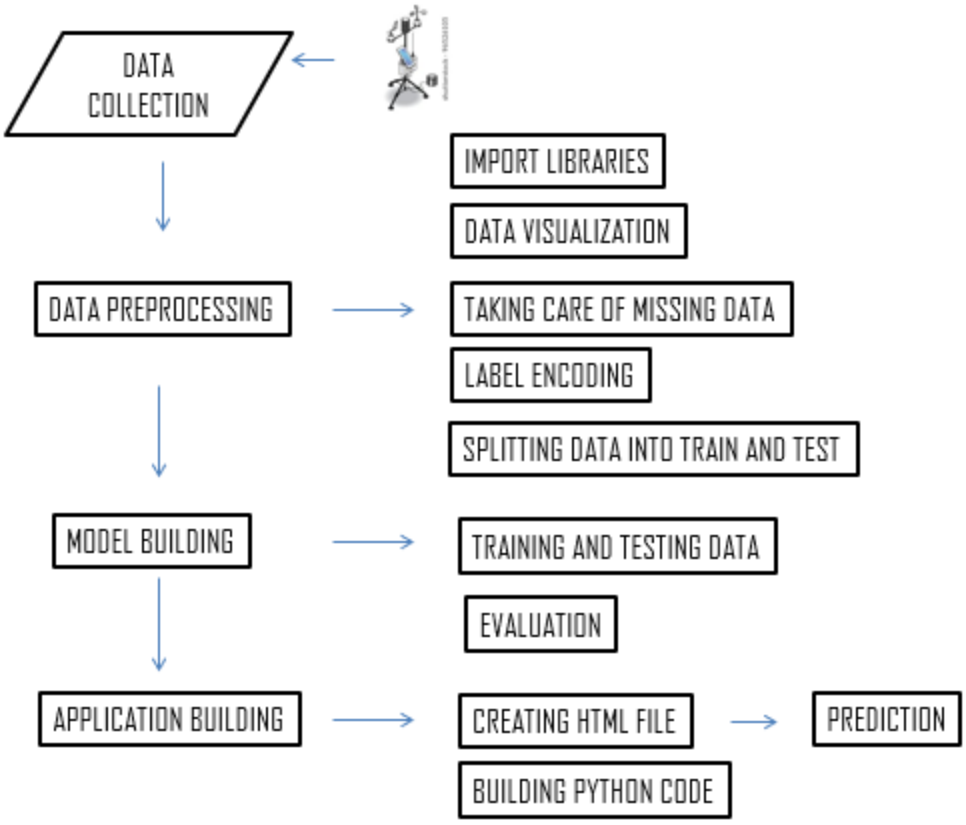
4. EXPERIMENTAL INVESTIGATION:

The avalanche is predicted with the given input level such as

1)Slope
2)Forest density
3)Snow density
4)Air temperature
5)Wind

```
In [29]: clf.predict([[34,1,51,-4,35]])
Out[29]: array([1], dtype=int64)
```

5. FLOW CHART:



6. RESULT:

We got an accuracy 0.95 which is a good measure for a Machine Learning algorithm used called Random Forest Classifier the model predicts the avalanche chances with good efficiency.

```
from sklearn import metrics
print("accuracy:",metrics.accuracy_score(y_test, y_pred))
```

```
accuracy: 0.9523809523809523
```

7. ADVANTAGE & DISADVANTAGE:

ADVANTAGE

1) Effective prediction of avalanche chances
2)Easily accessible web framework
3)Efficiency is maintained asap by updating data

DISADVANTAGES:

1) To predict the efficient output, necessary input is required
2) Updating the dataset is tedious process

8. APPLICATION:

1)Avalanche plays an important event during winter and spring.
2)The application using flask frame with the help of machine learning model Random forest classifier,with this application we can help may lives.

9.CONCLUSION:

Using Machine learning model we predict the chance of avalanche when the avalanche chance is low, moderate or high with required input

1) Slope
2) Forest density
3)Snow density
4)Air temperature
5) Wind

10. FUTURE SCOPE:

Using avalanche prediction we could predict time that at which time there is a chance of avalanche occur.

11.BIBILOGRAPHY:

•Dataset –kaggle
https://www.kaggle.com/aaroncareaga/avalanche-forecasting/data

12. APPENDIX:

12.1 DATASET:

	A	B	C	D	E	F
1	Slope	Forest De	Snow Den	Air Tempe	Wind	Prediction
2	59	H	52	-17	19	2
3	24	L	23	-8	19	0
4	15	H	30	7	19	0
5	27	L	61	-13	25	1
6	34	H	4	-20	7	0
7	34	L	76	-7	40	1
8	34	M	73	-32	27	2
9	34	L	4	3	9	1
10	34	M	44	6	6	0
11	34	L	73	-34	13	2
12	34	M	33	13	35	1
13	34	L	51	-4	35	1
14	34	M	59	-29	8	2
15	34	L	64	-21	10	2
16	34	L	76	-27	38	2
17	34	H	2	-23	33	2
18	75	L	70	-11	36	2
19	36	H	43	2	37	1
20	48	L	55	15	19	1
21	29	M	11	-25	38	0
22	59	H	27	-27	6	1

12.2 PRE-PROCESSING DATA

Out[22]:

	Slope	Forest Density	Snow Density	Air Temperature	Wind	Prediction
0	59	0	52	-17	19	2
1	24	1	23	-8	19	0
2	15	0	30	7	19	0
3	27	1	61	-13	25	1
4	34	0	4	-20	7	0
...	...	...	...	...	...	...
199	34	2	44	6	6	0
200	34	1	73	-34	13	2
201	34	2	33	13	35	1
202	34	1	51	-4	35	1
203	34	2	59	-29	8	2

204 rows × 6 columns

12.3 DATAPREPROCESSING CODE:

```
In [3]: from sklearn.preprocessing import LabelEncoder
        label_encoder = LabelEncoder()
        dataset['Forest Density'] = label_encoder.fit_transform(dataset["Forest Density"])
        dataset
```

```
In [4]: x = dataset.iloc[:,0:5].values
        y = dataset.iloc[:,-1].values
```

```
In [7]: import seaborn as sns
        sns.boxplot(dataset['Slope'])
```

```
In [13]: dataset.isnull().any()
```

```
Out[13]: Slope               False
         Forest Density      False
         Snow Density        False
         Air Temperature     False
         Wind                False
         Prediction          False
         dtype: bool
```

```
In [14]: from sklearn.model_selection import train_test_split
         x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.1,random_state=0)
```

12.4 MODEL TRAINING:

In [21]:
```
from sklearn.ensemble import RandomForestClassifier


clf=RandomForestClassifier(n_estimators=1500)


clf.fit(x_train,y_train)

y_pred=clf.predict(x_test)
```

In [23]:
```
from sklearn import metrics
print("accuracy:",metrics.accuracy_score(y_test, y_pred))
```

```
accuracy: 0.9523809523809523
```

In [24]:
```
from joblib import dump
dump(clf,"Random1.save")
```

Out[24]:
```
['Random1.save']
```

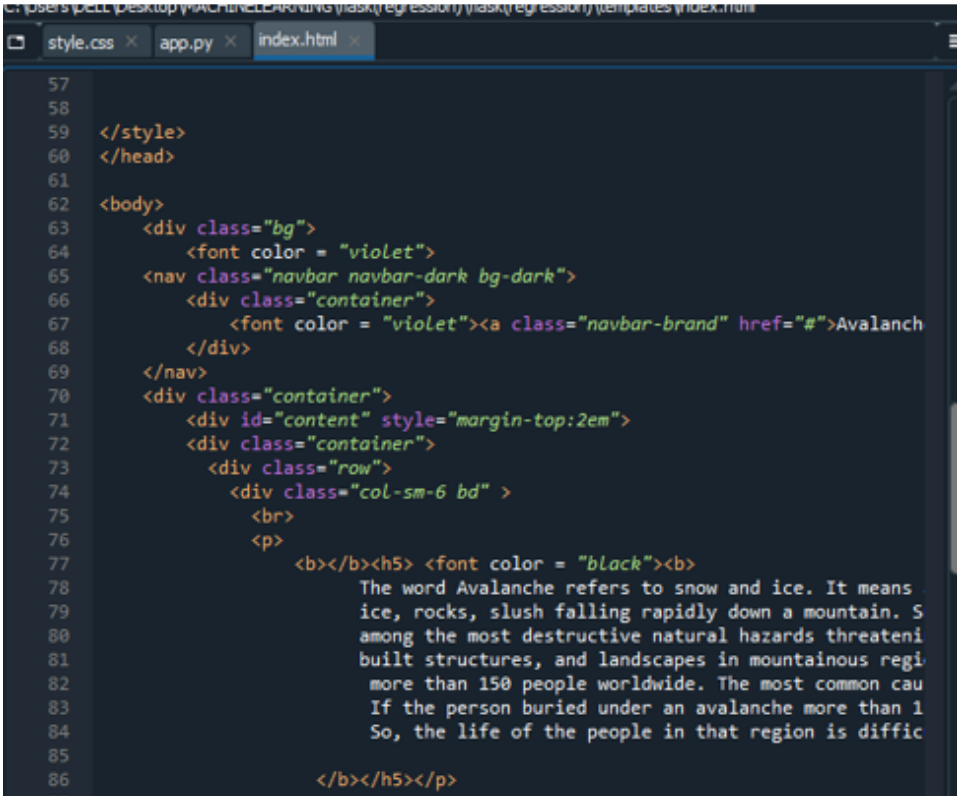
12.5 FLASK:

```python
import numpy as np
from flask import Flask, request, jsonify, render_template
from joblib import load
app = Flask(__name__)
model=load("Random1.save")
@app.route('/')
def home():
    return render_template('index.html')

@app.route('/y_predict',methods=['POST'])
def y_predict():
    '''
    For rendering results on HTML GUI
    '''
    x_test = [[int(x) for x in request.form.values()]]
    print(x_test)

    prediction = model.predict(x_test)
    output=int(prediction[0])
    print(prediction)
    if(output==0):
        a=" Low"
    elif(output==1):
        a="Moderate"
    else:
        a="High"

    return render_template('index.html', prediction_text='The Avalanche wil
```

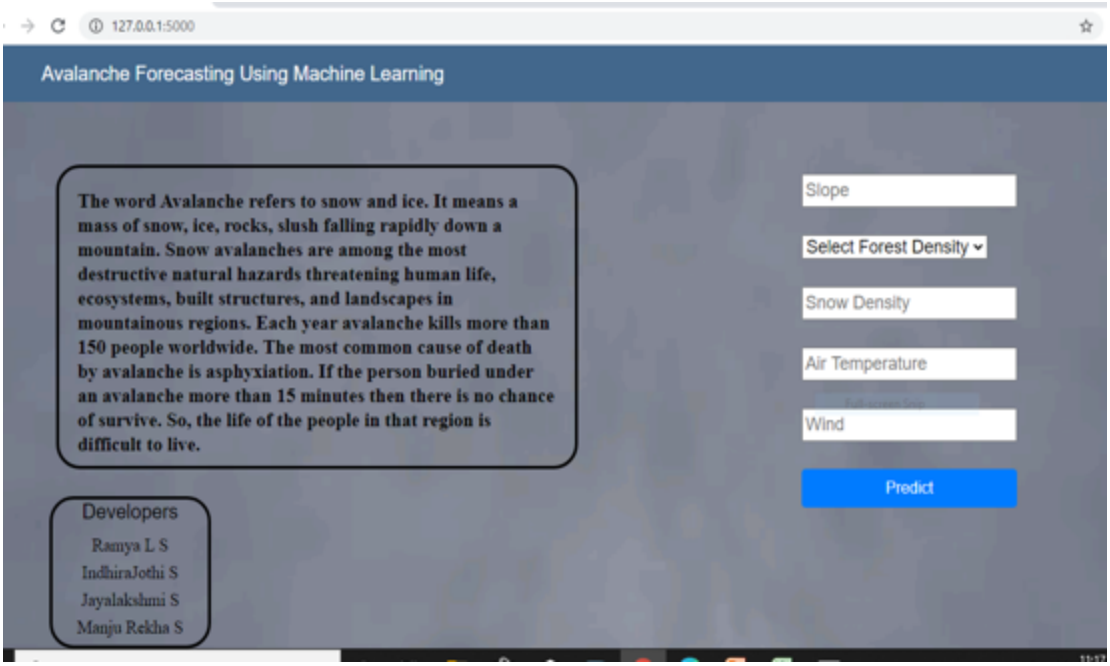



```
</style>
</head>

<body>
    <div class="bg">
        <font color = "violet">
    <nav class="navbar navbar-dark bg-dark">
        <div class="container">
            <font color = "violet"><a class="navbar-brand" href="#">Avalanch
        </div>
    </nav>
    <div class="container">
        <div id="content" style="margin-top:2em">
        <div class="container">
          <div class="row">
            <div class="col-sm-6 bd" >
              <br>
              <p>
                  <b></b><h5> <font color = "black"><b>
                        The word Avalanche refers to snow and ice. It means
                        ice, rocks, slush falling rapidly down a mountain. S
                        among the most destructive natural hazards threateni
                        built structures, and landscapes in mountainous regi
                         more than 150 people worldwide. The most common cau
                         If the person buried under an avalanche more than 1
                         So, the life of the people in that region is diffic

                    </b></h5></p>
```

BEFORE PREDICTION:



AFTER PREDICTION: