Machine Learning based Earthquake Early Warning (EEW) System: A case study of Himalayan Region

Samik Basu*[1], Arkadip Maitra[2], Soumen Halder[3], Soumya Pandit[4], Soma Barman (Mandal)[5], Pritha Banerjee[6] and Amlan Chakrabarti[7]

*corresponding author

[1,4,5] Institute of Radio Physics and Electronics, University of Calcutta, India

sbasurpe@caluniv.ac.in

sprpe@caluniv.ac.in

sbrpe@caluniv.ac.in

[2,3] Department of Computer Science, Ramakrishna Mission Vivekananda Educational and Research Institute, India

arkadip@rkmvccrahara.ac.in

soumen100@rkmvccrahara.org

[6] Department of Computer Science and Engineering, University of Calcutta, India

prithabcomp@caluniv.ac.in

[7]A.K. Choudhury School of Information Technology, University of Calcutta, India

acakcs@caluniv.ac.in

Abstract- Seismic sensing and generation of earthquake alarm is an important application for society at large. In this paper, we propose the strategy of extracting earthquake event features parameters τ_c *and* P_d *from fast-arriving P-wave signals. The said features are used to explore the performances of some of the popular machine learning (ML) based classifiers to compare their performances in triggering an alarm for the Earthquake Early Warning (EEW) system. We explored four different ML classifiers namely Support Vector Machine (SVM), Naive Bayes, K-Nearest Neighbors (KNN), and Logistic Regression so that the best can be applied for the EEW alarm generation. We have used publicly available data from the PESMOS platform of IIT-Roorkee in this work.*

Keywords - earthquake early warning, machine learning algorithm, magnitude, P_d, STA-LTA, τ_c

I. INTRODUCTION

Earthquakes are unwanted and unfavorable for the entire world. It hampers human life and property a lot. So an efficient warning system can help to avoid such circumstances. Earthquake early warning (EEW) system would help in the rapid detection of earthquakes, real-time assessment of the shaking hazard, and notification of people before shaking happens [1]. The range of warning time varies from a couple of seconds to handful of minutes depending on the scale and location of the earthquake. The further the location from the epicenter, there is more time to generate the warning. The seconds to minutes of advanced warning can enable people and systems to take required actions to safeguard life and property from destructive shaking [2].

The seismic waves contain primary wave (P-wave), secondary wave (S-wave), and surface waves (e.g. L-wave/R-wave) are radiated outward from the epicenter when an earthquake occurs [3]. The P-wave moves fast and is not destructive but it is crucial to be detected for onsite early warning purposes. In the onsite early warning approach, we mainly observe the first signs of the P-wave or called event detection and from there we predict the ensuing ground motion at the same site. [4] Two approaches are adopted for EEW System (a) Regional Warning and (b) Onsite Warning.

(a) Regional Warning: In seismological methods that are generally available the data from a specific seismic network is analyzed to detect the origin, find out magnitude and approximate ground motions of the impacted region. This warning approach usually takes extended instance and is not able to generate early warning when close to the epicenter [5].

(b) Onsite Warning: The initial portion of the ground motion (mainly P-wave) detected at a location is utilized to forecast the ensuring seismic activity (mainly S-wave and surface wave) at that region. The onsite method is quicker and provides essential early warning to locations at a short distance from epicenter, where the early warning is very essential [5].

Various researchers have created various systems to predict the magnitude of earthquakes using the EEW system parameters. The development of earthquake early warning system for Kachchh, Gujarat has used EEW system parameter τ_c and P_d for detecting the magnitude [6]. The multi-parameter algorithms for earthquake early warning developed by the data from the K-NET seismic array (Japan) have also determined the relationship between magnitude and the parameters [7]. Bhardwaj et.al. predicted τ_c, P_d and magnitude (M_w) relationship and the goodness of fit of the regressions in the Indian context [8]. 'Development of an earthquake early warning system using real-time strong motion signals' by Yih-Min Wu, Hiroo Kanamori that utilized τ_c, P_d as viable parameter for detecting earthquake [9].

To the best of our knowledge, the existing research approaches used regression techniques for predicting the earthquake magnitude. We believe that, formulating this as a classification problem is a better approach, where we labeled the outcome to generate alarm or not generate alarm. Our motivation is creating an earthquake alarming system primarily targeted towards users. The usability of this system is high as users can get easily warned with an alarm. We propose an ML-based earthquake alarming system using the EEW system parameters τ_c and P_d. Our design aims to develop a software model to trigger an alarm as an early warning system for an upcoming earthquake event. Some of the most commonly available classifiers were compared to see the difference in performance. We explored four different ML based classifiers and compare among them to get the best accuracy of results in terms of performance. Another objective of this paper is to reduce the number of false alarm and it was easily achieved due to the nature of the dataset.

The remaining part of the paper is organized as follows. In Section II we discuss data collection and preparation. In section III we describe an event detection algorithm using STA/LTA and in Section IV we discuss EEW system parameters. Section V contains the proposed workflow and Section VI briefs the methodology, which demonstrates different machine learning algorithms. Section VII contains the result and discussion. We conclude this paper in Section VIII.

II. DATA COLLECTION AND PREPARATION

Data preparation is a major paradigm for this work. We started data collection and preparation of earthquake data which was downloaded from the PESMOS (2005 to 2014) site [10]. At the time of the data processing, we thoroughly followed the data science life cycle [11]. But before any processing, we are required to understand data properly. PESMOS data contains three components that are: (East-West-EW, North-South-NS, and Vertical).

```
Origin Time       27/06/2008 11:40:16
Lat.              11.0 N
Long.             91.6 E
Depth (Km)         10.0
Magnitude         6.7
 Region           Andaman
 Above details taken from IMD

Station Code      POR
Station Lat.       11.664
Station Long.      92.742
Station Height(m)    5.0
Site Class        B   Vs30 between 375 m/sec to 700 m/sec *
Record Time       27.06.2008 11:40:24.494
Sampling Rate     200. Hz
 Record Duration   63.410 Sec.
 Direction         Vert. (Up positive)
Max. Acceleration    -22.505 cm/sec**2

  * For reference see Site Classification link of website

 Base Line Corrected and Low Pass Filtered (Cut Off at 35 Hz) Time History
 Acceleration data in cm/sec**2

      -0.130
      -0.029
       0.021
       0.026
       0.014
       0.010
       0.016
       0.021
       0.015
       0.003
      -0.004
      -0.002
       0.003
       0.003
       0.001
       0.003
       0.010
       0.016
       0.014
       0.004
```

Figure 1: Downloaded data from PESMOS Site [9]

In this present work, we consider vertical component data of content "Origin, Time, Latitude, Longitude, Depth (Km), Magnitude, Acceleration" with station details. Figure 1. shows an example of downloaded data from the PESMOS site [10]. It shows all details of the data. Now we had raw data and needed to work with missing values. We followed average value techniques to fill the data. We have prepared, cleaned and made the data readable for further processing.

III. EVENT DETECTION ALGORITHM

In this paper, we have applied an event detection algorithm for P-wave event detection. We use the Short-Term Averaging / Long-Term Averaging (STA/LTA) algorithm [12] for the same.

Short Term Averaging (STA) window: STA measures amplitude of particular moment from seismic signal. In the present work, the point number in the STA window is set at 5.

Long Term Averaging (LTA) window: This is used to take care of average seismic noise. The point number in the LTA window is set at 120.

The STA/LTA ratio signifies the signal-to-noise ratio (SNR).

STA/LTA Algorithm: The STA/LTA algorithm constantly logs the changes in seismic noise amplitude that happens at particular station site [13]. It is constantly compared with a pre-selected level of threshold. If the ratio is higher or equal to a pre-defined user-selected threshold level [12], then the event is detected and a trigger occurs.

$\mathrm{STA}(i) = \frac{1}{n_s}\sum_{j=i-n_s}^{i} S_j$ short-term averaging (1)

$\mathrm{LTA}(i) = \frac{1}{n_l}\sum_{j=i-n_l}^{i} S_j$ long-term averaging (2)

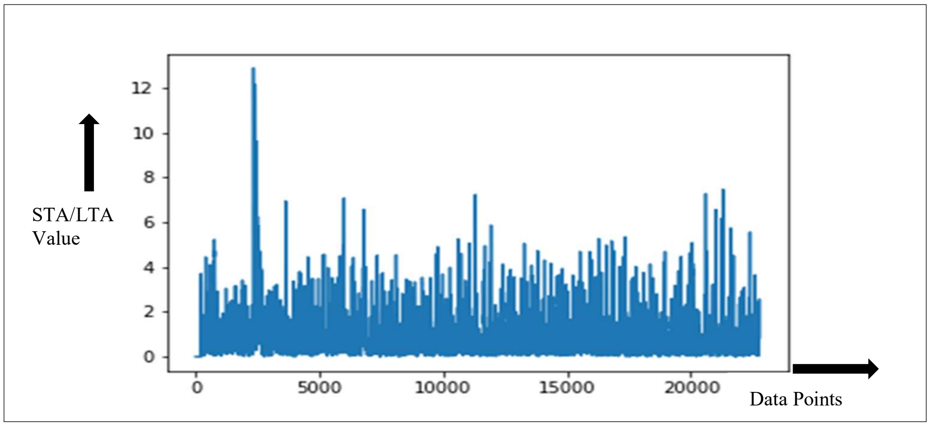


Figure 2. STA/LTA Plot

The ratio of STA/LTA is given by:

$$\frac{\mathrm{STA(i)}}{\mathrm{LTA(i)}} = \frac{\sum_{j=i-ns}^{i} h(j)S(i-j)}{\sum_{j=i-n}^{i} g(j)S(i-j)} = r(i) \qquad (3)$$

$h(j) = 1/n_s$ and $g(j) = 1/n_l$ are constant, where h(j) and g(j) are sample response unit and measure of all the coefficients of h(j) and g(j) are constant. If r(i) is greater than or equal to a predefined user-selected threshold value then the event is detected and a trigger occurs. The STA/LTA plot is shown in Figure. 2. This signal-to-noise ratio (SNR) represented by STA/LTA algorithm is continuously measured against a pre-determined level of threshold. We observed from Figure. 2. that the first major spike is detected (an event is detected) using the STA/LTA algorithm. Here we set a user-selected predefined threshold value of 9[13]. Thus, the spike noted around 2500 on the x-axis denotes the arrival of the P-wave event with a value of more than 12 (greater than the user-selected threshold value of 9) on the y-axis.

IV. EEW SYSTEM PARAMETER

We need to evaluate two important EEW system parameters that are τ_c and P_d. Kanamori (2005) has extended the work method of Nakamura (1988) and R. Allen, H. Kanamori (2003) to determine the parameter τ_c [5,14,15]. The average time period of an initial portion (3-5sec) of P-wave is defined as

$$\tau_c = 2\pi\left(\frac{\sqrt{\int_0^{t0} u^2(t)dt}}{\sqrt{\int_0^{t0} u'^2(t)dt}}\right) \qquad (4)$$

where u' and u are velocity and displacement respectively. The integration is taken over a time interval $(0, t_0)$ after the onset of the P-wave. Usually, t0 is set to be 3 seconds. The maximum displacement within 3 seconds after the arrival of the P-wave, i.e., P_d [5,15] is very crucial and is found from acceleration data through double integration. The mathematical expression of velocity from discrete value acceleration based on cumulative trapezoidal rule is given by

$$u'_n = \frac{1}{2}[(u''_n) + 2(u''_0 + u''_1 + \ldots + u''_{n-1})] \qquad (5)$$

Similarly, mathematical expression of displacement from discrete value velocity based on cumulative trapezoidal rule is given by

$$u_n = \frac{1}{2}[(u'_n) + 2(u'_0 + u'_1 + \ldots + u'_{n-1})] \qquad (6)$$

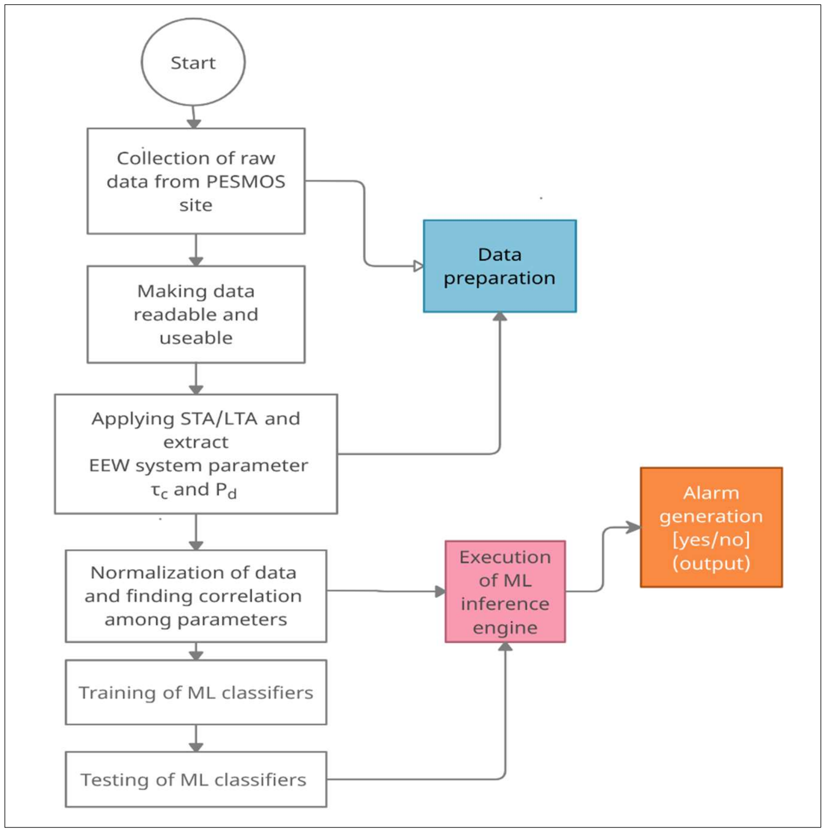


Figure 3: Workflow Model

V. PROPOSED WORKFLOW

A. An Overview

The proposed system is developed using a simple workflow model shown in Figure 3. The vertical component of the PESMOS data is first cleaned, missing values are then filled up and given for computing the EEW system parameters τ_c and P_d. The EEW parameters (τ_c and P_d) are then computed from the processed vertical component data of each station using the STA/LTA event detection algorithm described above. The EEW parameters derived for each station (after the event is detected) are then normalized using z-score normalization [16]. Correlation among parameters i.e. τ_c, P_d, and M_w are checked as a high correlation can result in poor accuracy in output. Here, the Pearson correlation coefficient [17] is used. The normalized τ_c and P_d data are then being used for classification using machine learning-based (ML-based) classification algorithms namely Support Vector Machine (SVM), Naive Bayes (NB), K-Nearest Neighbor (KNN), and Logistic Regression (LR).

B. Implementation of STA/LTA on Cleaned Prepared Data

From the raw dataset the vertical component data have been taken and then it was preprocessed as follows:

- Calculation of STA/LTA - Implementing the stated Event detection algorithm using STA/LTA, setting the short-term window as 5 and long-term window as 120.
- Calculating τ_c and P_d- The STA/LTA ratio is the signal-to-noise ratio and the value of it is continuously compared with a user-defined threshold value. The threshold value is selected based on the past seismic record of a particular site. In the present work, we have taken STA/LTA threshold value as 9 to calculate τ_c and P_d [13].

- Arranging the data- All the data is then organized in a dataset consisting of the station, τ_c, and P_d as columns. Magnitude (Mw) is known and downloaded from the PESMOS site [10].
- Python code has been used for extracting the value of τ_c and P_d (EEW system parameters).

C. Obtaining EEW System Parameters Value

Table 1 is obtained by calculating the τ_c and P_d values of each station after an event is detected and the magnitudes here, have mentioned are known to the user.

D. Calculating Z-Score Value

Z- score [16] is used to describe an observation of data point being far from average, more technically how many standard deviations occur above or below from the mean value of the population. In machine learning, normalization ensures that the feature space of the dataset is well represented on a common scale without distortion of the range of values. To use a z-score, we need to know the value of x, the mean μ, and also the population standard deviation σ. The formula of z- score for a sample is given by: $z = (x - \mu) / \sigma$. The outcome is showed in Table 2.

E. Calculating Pearson Correlation Coefficient

The measurement of a linear correlation between two variable points or two sets of data represented by ɍ [17], which indicates how well the data points are fitting in a new model or the line of best fit. This is termed as Pearson correlation coefficient.

$$ɍ = \frac{\sum(p_i - p')(q_i - q')}{\sqrt{\sum(q_i - q')^2 \sum(p_i - p')^2}} \qquad (7)$$

ɍ = factor of correlation
p_i = p-variable value in sample
p' = mean of p-variable value
q_i = q-variable value in sample
q' = mean of q-variable value

Table 1: Data Sample of Required Parameters

Station	P_d (in cm)	τ_c (in s)	Magnitude (Mw)
20080529 DABRNG, ASSAM	0.014636356	3.749944716	4.2
20080810 ANUDAMAN ISLAND	0.827137419	5.75956837	6
20080529 DARRANG, ASSAM	0.007629244	2.173486637	4.2
20090810- ANDAMAN ISLAND	0.405102181	9.961815737	7.8
20100621-OFF WEST COAST OF	0.039869919	2.09408412	6.2

NICOBAR ISLAND			
20131020 CHINA BORDER	0.017485894	5.45873493	5.5
20090921 BUTAN	0.03989919	2.09408412	6.2

Table 2: Data Sample of Normalized P_d and τ_c

P_d (in cm)	**τ_c (in s)**
0.860289	1.149189
0.373016	-1.230609
-0.298465	-0.985851
-0.381215	-1.101757
-0.187594	-1.112672
1.552810	0.433146
-0.290098	-1.054427

Table 3: Correlation Among the Parameters

Correlation	**P_d**	**τ_c**	**Magnitude (Mw)**
P_d	1.000000	0.304642	0.347933
τ_c	0.304642	1.000000	0.235533
Magnitude (Mw)	0.347933	0.235533	1.000000

In Table 3, the correlation among P_d, τ_c, and Magnitude (Mw) is shown. It is observed from the table that the correlation among P_d, τ_c, and Magnitude (Mw) is well below its threshold value of 0.6 [18]. This ensures that the parameters P_d, τ_c is independent and will not impact each other.

VI. METHODOLOGY

We explore four different classifiers namely Support Vector Machine (SVM), Naive Bayes (NB), K-Nearest Neighbors (KNN), and Logistic Regression (LR) to classify an earthquake event for an early warning. These four classifiers are chosen to serve as the baseline classifiers for categorizing seismic magnitudes. Further classification techniques can be implemented in future and results can be compared with this baseline. The normalized form of the EEW system parameters is used in the classifiers. The two parameters τ_c and P_d are used as the X label of the classifiers. The magnitude (Mw) is used for defining class. The magnitude below or equal to 5.5 is denoted as class 0 and the magnitude above 5.5 is denoted as class 1. The class labels are treated as Y. The threshold value for generation of alarm is set at 5.5 because the severity of earthquakes is widely observed after a magnitude of 5.5 according to standard seismic scale [19]. The objective of preventing widespread destruction is therefore achieved by generating alarm after a magnitude of 5.5.

The accuracy metric is the rate at which a classifier has classified X into its correct class i.e. whether the magnitude is less or equal to 5.5 or the magnitude is greater than 5.5. Some other information like the number of times the classifier falsely denoted X as class 0 i.e. False-negative and also the number of times the classifier falsely denoted X as class1 i.e. False-positive. Our goal being the reduction of false alarms and the False-positive rate is the metric to determine that.

<u>Preprocessing steps followed before classification:</u>

1. The normalized P_d, τ_c are treated as X and the class data is treated as Y.
2. We consider two classes 1 and 0 based on Magnitude scaling.
3. Class 0 is for the instances having a Magnitude less or equal to 5.5 and Class 1 is for the instances whose Magnitude is greater than 5.5
4. We split data into an 80/20 ratio in the training and testing part for all the methods except the KNN.
5. Four ML based classifiers SVM, NB, KNN, and LR are applied.

a) Support Vector Machine (SVM): A support vector machine [20] is a supervised classification algorithm. SVM separates n-dimensional space into classes by making decision boundaries. A hyperplane is the best choice to separate data in classes. Support Vectors are known as data points or vectors which affect the hyperplane position most. SVM can be used for two types of decision boundary, Linear and Nonlinear. In this work, we used Nonlinear SVM. SVM uses one or more mathematical functions to derive the optimal decision boundary or a margin it termed as a kernel. To accommodates non-linear class boundary, the Radial Basis Function (RBF) kernel [21] is used in this work.

b) Naive Bayes (NB): A Naive Bayes [22] classifier is a supervised learning algorithm that uses probability theory to classify data points into separate classes. It assumes that each of the attributes of a data point under consideration is independent of each other, thus, denoted as naive. Given a problem instance to be classified, represented by a vector $X= \{x_1, x_2, \ldots, x_n\}$, it assigns to this instance probabilities $P(C_k \mid X)$ for each of the K possible classes C_k. The Naïve Bayes classifier then combines this model with a decision rule. Thus, the classifier assigns a class label $Y = C_k$ such that $P(C_k)\ \Pi\ P(x_i| C_k)$ is maximized [23].

c) K-Nearest Neighbors (KNN): K-Nearest Neighbors [24] is a supervised, non-parametric classification algorithm. KNN algorithm does not learn from the training phase but stores the dataset and when new data or test data comes, it classifies that data into a class by finding similarities between the new data point and the stored data set. The 'K' in KNN is the number of nearest neighbors considered in the algorithm. This algorithm starts with a smaller random value of 'K' and fixes the value using various standard processes like cross-validation, trial, and error, etc. In this algorithm when a new data point comes, it classifies it by a majority voting of the classes of K closest neighbors. In this work, we use K = 5. The distance metric used here is the Euclidean distance measure.

d) Logistic Regression (LR): Binary Logistic Regression [25] is a nonlinear supervised special type of regression algorithm used for classification purposes. This model is used to predict a given binary value (0/1 or 'yes / no') as a set of independent variables, where the binary variable can be discrete and/or continuous. The expected values of the response variable are modeled on the basis of a combination of values taken by the predictors.

Generally, it uses the sigmoid function as a discriminatory function. A Sigmoid function is defined as:

$$y = \frac{e^{(b_0 + b_1 * x_1 + b_2 * x_2)}}{(1 + e^{(b_0 + b_1 * x_1 + b_2 * x_2)})} \quad (8)$$

where b_0 represents the bias and b_1, b_2 are the coefficients of the given data points x_1 and x_2. The value of bias and coefficients are determined through training.

The hyperparameters for all the classifiers used are given in table 4. The gamma is used as a coefficient by the rbf kernel. The scale uses $\frac{1}{(\text{n\_features} * \text{X.var}())}$ as value of gamma.

The variance smoothing is the portion of that portion of the largest variance of all features added to variances. The weight parameter assigns weight to each neighbor. Uniform weight is assigned to each neighbor. The penalty parameter determines the type of regularization ie. l_1 or l_2. The solver is an optimization parameter.

VII. RESULT AND DISCUSSION

Table 5 illustrated the output plot of each of the classifiers along with the confusion matrix. The confusion matrix [26] for each of the classifiers is generated using the test data.

The performance of all the classifiers is compared and shown in Table 6. As evident from the confusion matrix outcome, the KNN classifier has the best performance followed by LR, SVM, and NB classifier as showed in Table 6.

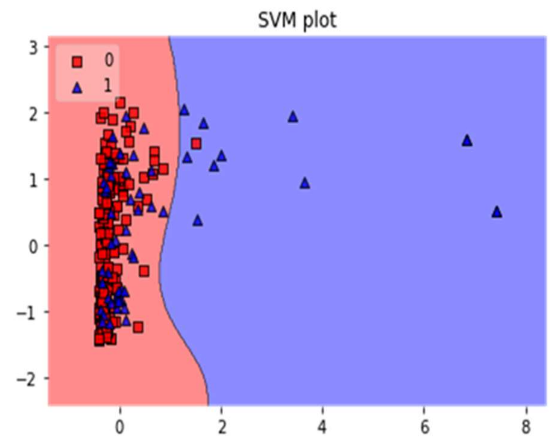
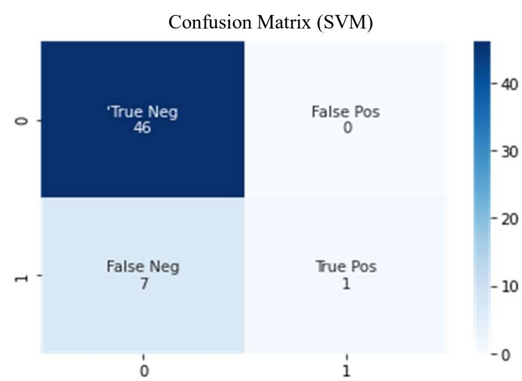
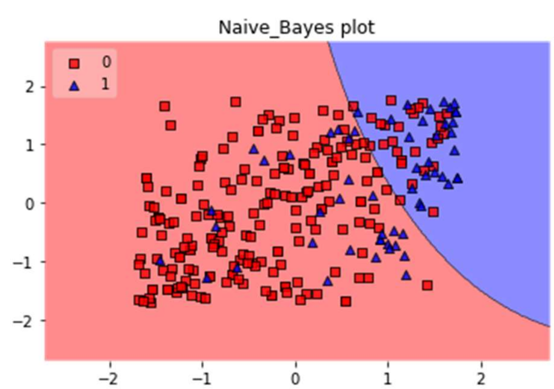
Table 4: Table of Hyperparameters

Classifier	**Hyperparameters**
Support Vector Machine(SVM)	Regularization parameter = 1 Kernel = rbf Gamma = scale
Naïve Bayes(NB)	Var smoothing = $1e^{-9}$
K Nearest Neighbors(KNN)	Neighbor numbers = 5 Weight = uniform
Logistic Regression(LR)	Penalty = l_2 Solver = lib linear

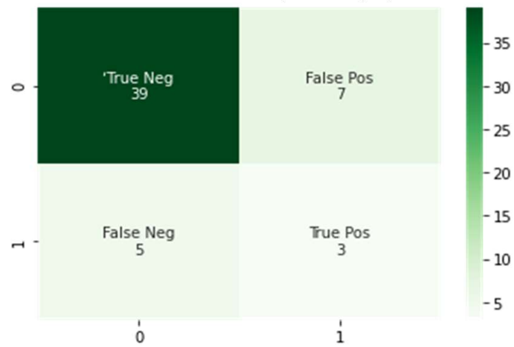
The precision, recall, and F_1 score gives further information beyond the accuracy of the classifier.

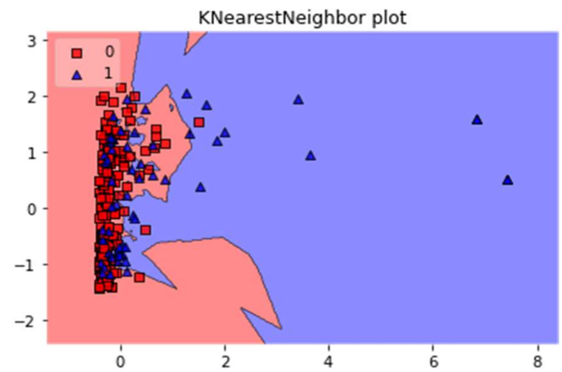
We have summarized that the KNN classifier has given the best performance as it provides the highest accuracy along with the lowest false positive and false negative rate to trigger an alarm for an upcoming earthquake event. The KNN classifier also has the highest precision, recall, and F_1 score which made it the best choice.

Table 5: Classification Results

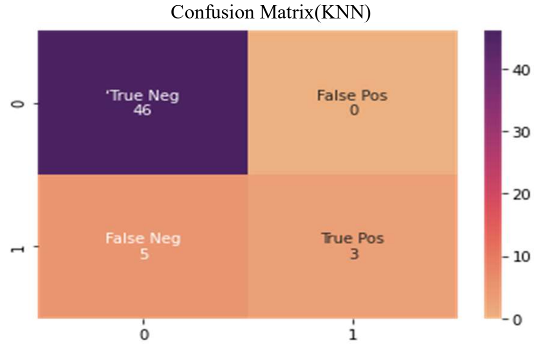
Algorithms	**Plots**
Support Vector Machine (SVM)	 
Naive Bayes (NB)	

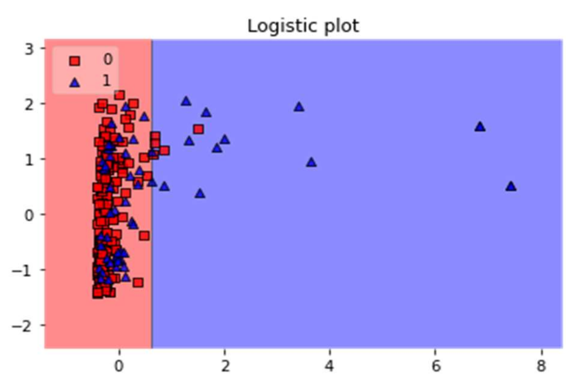
<table>
<tr>
<td></td>
<td>
Confusion Matrix (Naive Bayes)



</td>
</tr>
<tr>
<td>K Nearest Neighbors (KNN)</td>
<td>


Confusion Matrix(KNN)



</td>
</tr>
<tr>
<td>Logistic Regression (LR)</td>
<td>


</td>
</tr>
</table>

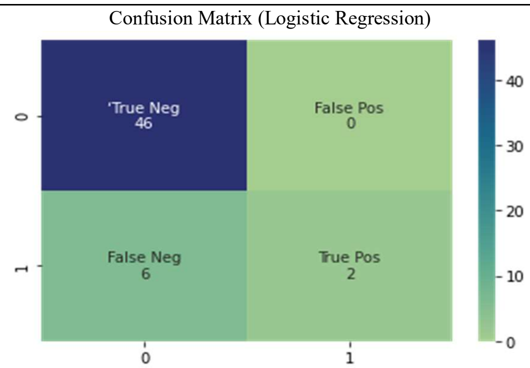


Table 6: Comparison of Results

Classifier	Accuracy in %	False Positive in %	False Negative in %	Precision	Recall	F_1 Score
SVM	87%	0%	13%	1	0.125	0.222
NB	78%	13%	9%	0.3	0.375	0.333
KNN	91%	0%	9%	1	0.375	0.545
LR	89%	0%	11%	1	0.25	0.4

VIII. CONCLUSION

This work showed that seismic data of the Indo-Himalayan region can be well classified to determine the severity of an earthquake and the need for raising an alarm. The comparison among the performance of the various classifiers allowed us to choose the highest accuracy classifier which can provide the lowest failure rate. We are planning to implement a better classification system with larger dataset in future. We will also implement this design to FPGA-based hardware to exploit its hardware acceleration feature. In the future, we also plan to evaluate the real-time performance of the embedded classifier system.

REFERENCES

[1] Berkeley Seismology Lab. "http://seismo.berkeley.edu"

[2] USGS: "https://www.usgs.gov/natural-hazards/earthquake-hazards/early-warning?"

[3] Seismic Education Site - UPSeis: http://www.geo.mtu.edu/UPSeis

[4] Samik Basu, Soumya Pandit, Amlan Chakrabarti, Soma Barman (Mandal), "FPGA based Hardware Design for Noise Suppression and Seismic Event Detection", proceedings of 5th IEEE International Symposium on Smart Electronic Systems (iSES), IEEE-iSES 2019, 16-18 December 2019, NIT Rourkela, India, ISBN: 978-1-7281-4655-3, DOI: 10.1109/iSES47678.2019.00094

[5] H. Kanamori, "Real-time Seismology and Earthquake Damage Mitigation", Annual Review of Earth and Planetary Sciences, 19 May 2005, Vol. 33:195-214

[6] Santosh Kumar, “Development of earthquake early warning system for Kachchh, Gujarat, in India using τc and Pd” Arabian Journal of Geosciences, July 2020, DOI: 10.1007/s12517-020-05353-3

[7] R. Bhardwaj, “Multi-parameter algorithm for Earthquake Early Warning, Geomatics, Natural Hazards and Risk”, 1242-1264, DOI: 10.1080/19475705.2015.1069409.

[8] R. Bhardwaj, “Analysis of Tauc (τc) and Pd attributes for Earthquake Early Warning in India”, 15th World Conference on Earthquake Engineering, 2012, 4068-4075.

[9] Yih-Min,” Development of an Earthquake Early Warning System Using Real-Time Strong Motion Signals” Sensors (Basel), 2008 Jan 9;8(1):1-9. doi: 10.3390/s8010001.

[10] PESMOS, Department of Earthquake Engineering, IIT-Roorkee. https://www.pesmos.in”.

[11] Sumeet Bansal, “Data Science Life Cycle”, Analytixlabs, 2017.

[12] S.Gaci, “The Use of Wavelet-Based Denoising Techniques to Enhance the First-Arrival Picking on Seismic Traces”, IEEE Trans. on Geoscience and Remote sensing, 2014,vol. 52, no.8.

[13] A. Trnkoczy, “Understanding and parameter setting of STA/LTA trigger algorithm”, GFZ publication database, September 1999, DOI:10.2312/GFZ.NMSOP_R1_IS_8.1.

[14] R. Allen, “The potential for earthquake early warning in southern California”, Science, 2003 May 2; 300(5620):786-9. doi: 10.1126/science.1080912. PMID: 12730599.

[15] Wu Yih-Min, “Determination of earthquake early warning parameters, τc and Pd , for southern California”, Geophysical Journal International 170, 2007, 711-717.

[16] S Patro, “Normalization: A Preprocessing Stage”, Arxiv, 2015, 1503.06462.

[17] Eric Weisstein, "Correlation Coefficient." Wolfarm MathWorld.

[18] Carsten F. Dormann, “Collinearity: a review of methods to deal with it and a simulation study evaluating their performance”, Ecography, 2012, 35: 001–020, doi: 10.1111/j.1600-0587.2012.07348.

[19] Swiss Seismological Service, “Dealing with natural hazards – earthquakes danger levels” European Macroseismic Scale, 1998.

[20] S. Suthaharan, “Support Vector Machine. In: Machine Learning Models and Algorithms for Big Data Classification”, Journal of Data Analysis and Information Processing, pp. 207-235.

[21] GL. Prajapati, “On performing classification using SVM with radial basis and polynomial kernel functions”, 3rd International Conference on Emerging Trends in Engineering and Technology, November 2010 IEEE (pp. 512-515).

[22] G. I. Webb, “Naive Bayes”, Encyclopedia of machine learning, 2010, 713-714.

[23] Seyoung Kim, “Naïve BayesClassifier”, Machine Learning 10-601B”

[24] Leif E. Peterson, “K-nearest neighbor” Scholarpedia, 2009, 4(2), 1883.

[25] D.W. Hosmer Jr, “Applied logistic regression”, 2013, John Wiley & Sons, Vol 398.

[26] Sarang Narkhede, “Confusion Matrix”, Towards Data Science, 2018.