SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM
TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE
Volume 5 No. 5
ISSN 2054-7390

Urban Flood Forecast using Machine Learning on Real Time Sensor Data

[1] Likith Ponnanna P B, [2] R Bhakthavathsalam, [3]K Vishruth
[1,2]Supercomputer Education and Research Centre, Indian Institute of Science, Bangalore, India;
[3]College of Computer and Information Science, Northeastern University, Boston, USA
likith.april13@gmail.com; bhaktha@serc.iisc.in; krishnaprasad.v@husky.neu.edu

ABSTRACT

All the underpasses, flyovers and drainage networks in the urban areas are designed to manage a maximum rainfall. This situation implies an accepted flood risk for any greater rainfall event. This threat is very often underestimated as components such as climate change is disregarded. But even great structural alterations cannot assure that urban flood control precautions would be able to cope with all future rainfall events. Hence, being readily able to forecast city or urban floods in real time is one of the main tasks of this forecast. The current Urban flood forecasting methods involve the use of Geographical Information Systems techniques. Even though, these systems allow to detect and model the flood patterns in a larger perspective. They cannot pin point precise location behavior. Machine Learning models in conjunction with a sensor network can be essential elements of urban flood forecast systems, as an active part of the system or as study tools. The paper goes into the application of machine learning models to better predict flood pattern based on several external factors in real time.

Keywords: Urban Flooding, Flood Forecasting, Machine Learning, Real time Machine Learning.

1 Introduction

As the cities grow at a rapid pace, the city planning usually tends to fall behind. Extreme amount of concretization of these cities has brought up huge issues in the cities flood drains. These issues will be increasing as the concretization increases. Since a huge amount of time and investment is required to make better drainage systems some of the government bodies tend to ignore the issue until it becomes too big to handle. This can cripple most cities. As a result of this, a lot of the poorly planned cities all over the world have huge urban flooding problems. Where even a small rainfall brings the city traffic to a halt. And a larger rainfall causes the cities to come to a complete standstill. Which can be seen in the cases of 25 July 2005 in the city of Mumbai. The government surely would look into the issue had there been an economical alternative to detect, forecast the urban floods [1]. This, in turn, helps the government body to plan strategically and avoid unnecessary spending for projects with lesser overall effect on the flooding patterns, which drives down the cost of planning and implementation by placing the sensors in strategic locations.

The majority of the current solution is based on satellite data [2] using 3D Modelling techniques. Although these solutions have a considerable ability to detect flooding at a bigger area such as a completely

DOI: 10.14738/tmlai.55.3552
Publication Date: 20th August, 2017
URL: http://dx.doi.org/10.14738/tmlai.55.3552

submerged area, they cannot be used for mapping highly localized flooding such as a flyover or an underpass. Also, having these data collections at a highly localized level can help in detecting and fixing fault areas and could help in times where major disaster could be avoided by proper planning, as seen in the case of Chennai [3], India 2015 floods.

2 Setup

The hardware setup comprises of multiple sensors whose readings are handled by a micro-controller in real-time and pushes it to a hosted database. The different sensors [4] are as follows:

2.1 Ultrasonic Sensor

The ultrasonic sensor takes real-time readings of water level over time. The ping generations of the sensor are increased when the water level is found to be increased so that more granular data is available for the modelling. The ultrasonic sensor calculates the distance between itself and the ground. The initial distance and measurements are Pre-calibrated. The trigger part of the sensor bounces ultrasonic waves to the ground surface and waits to receive the echo back. Based on the time taken for the ultrasonic waves to travel back and forth the distance is measured which gets altered in case of a water level rise.

2.2 Thermometer

The thermometer is used to consider the temperature variations that are caused during different scenarios such as rain, heat, also helps to indicate if the flood was formed locally or was it carried from a different source. This helps in checking the slight temperature variations in conjunction with other variables in case of rain, snow, mist, etc., which might affect the precipitation values.

2.3 Barometer

A barometer is used to detect the atmospheric pressure. There is a large amount of correlation between atmospheric pressure and rainfall in a region [5]. Especially in times of cyclones or hurricanes if a very low-pressure area is developed the imminent rain can be sensed beforehand. It helps detect the possibilities of flooding in response to the rainfall.

2.4 Hygrometer

Hygrometer will work in conjunction with the other sensors to effectively monitor rainfall and other correlations driven from the humidity readings. Usually, there is a steep increase in the humidity at a location when there is rainfall.

2.5 Precipitation Sensor

The precipitation sensor is used to measure the rain intensity at any given time. The rain intensity gives an additional dimension when it comes to forecasting the water settlement in areas. In places where other precipitation parameters such as snow and mist are lower, a rain gauge can be used in its place to just monitor the rainfall intensity over time.

2.6 Ammeter paired to Solar Panel

Since cloud cover and day-night cycles affect the production of electricity by a solar panel. An ammeter is placed in conjunction with the solar panel to measure the intensity of the current, which helps in mapping the cloud cover and night cycles.

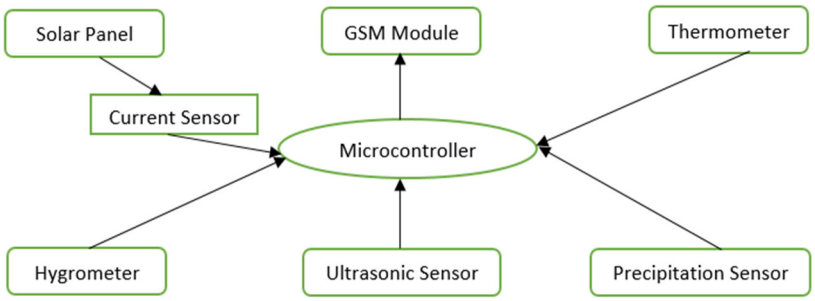

Figure 1. Hardware Setup

3 Methodology

3.1 Dimensionality Reduction

After the features required for the forecast is carefully chosen based on weather data relations. It is observed that the data is in a high dimensional space. To reduce this higher dimensional space to tangible few dimensions, feature extraction techniques are used. As we see the data is taken in real time the data scale is ever growing. For larger dataset, Linear Discriminant Analysis (LDA) is a much more suitable dimensionality reduction technique when compared to Principal Component Analysis (PCA) [6]. From *n* independent variables in the dataset, LDA extracts a more manageable number of new independent variables that separate the most, the classes of dependent variables.

Since there are more than two classes and the LDA is generalized from the Fisher discriminant. The Fisher discriminant could be extended to detect a subspace which encompasses all the class divergences. If each of the C classes has a mean μ_i and the same covariance Σ. The dispersion in between class variability may be outlined by the sample co-variance of the class means [7]

$$\Sigma_b = \frac{1}{c}\sum_{i=1}^{c}[\mu_i - \mu][\mu_i - \mu]^T$$

Where μ - average of class means. The class divergence in a particular direction $\vec{\omega}$ is provided by

$$s = \frac{\vec{\omega}\sum_b^T \vec{\omega}}{\vec{\omega}^T \Sigma \vec{\omega}}$$

$\vec{\omega}$ is an eigenvector of $\Sigma^{-1}\Sigma_b$ separation.

3.2 Classification Techniques

The dataset is run with multiple types of classification techniques which help in detecting the accuracy of different models. Using the accuracy check we can finalize a particular technique to maximize correct predictions.

3.2.1 K-nearest Neighbors(k-NN)

The k-NN is a classification technique, that is a nonparametric method and it makes its predictions based on the target outputs of its k nearest neighbors of any given point to be queried. Euclidean distance is calculated between every point and the training set points. The closest k training points are selected and prediction is taken as the average of the target output values of those k-points [8].

3.2.2 Kernel SVM

Kernel methods are used for pattern analysis. One of their members is support vector machines (SVM). The kernel methods utilize kernel functions to ply in higher dimensional and implicit area and without ever evaluating coordinates of the different data points. It in place uses internal products in between the images of all the data pairs in the feature or characteristic space [9].

3.2.3 Decision Tree Classification

This classification allows to build classification model based on the tree structure. It initially breaks the dataset into smaller subsets and in conjunction an associated decision tree inbuilt step by step. The result obtained is a tree with decision and leaf nodes [10].

3.2.4 Random Forest Classification

The Random Forest Classification develops lots of decision tree based on random selection of data and variables. The class of the dependent variable is based on multiple trees [11].

4 Implementation

4.1 Dataset

Using the hardware setup given above data collection was carried out by multiple sensors in different parts of the city of Bengaluru. This Data stream was collected for a time frame of twelve months. The data was then taken and values with multiple duplicates were removed to avoid redundancy of multiple sensor data. The final dataset was taken out which comprised of 5800 data points. The independent variables in the data set include parameters such as temperature, humidity, atmospheric pressure, current value from the solar panel which is set to three different levels, Precipitation levels, Water level, the dependent variable which is if there is a possibility of a flood is divided into 3 states, namely no flood, light flooding, heavy flooding which are assigned particular values.

Table 1. Database Schema

Sensor ID	Current(mA)	Temperature (℃)	Humidity	Atm Pressure (mb)	Precipitation / Rain Intensity	Water Level (cm)	Flood

4.2 Classifier Process

The dataset obtained is taken and split into training and test dataset. Since the amount of relationship can only be obtained by a significant training set and 20-80 split of training and test data is made. LDA is applied to reduce the dimensionality of the dataset independent variables. Different model classifiers are modelled onto the same dataset and the best model is chosen. The constant real-time stream is taken by the model and the new prediction values are computed constantly based on the optimal model.

Algorithm 1. General algorithm for multiple classifiers

Given a dataset D consisting multiple feature variables making the model of higher dimension. n is the number of feature variables.
for i ← 1 **to** N **do**
Read the dataset and construct dataset matrix
end for
for I ← 1 **to** N **do**
Extract the dependent and independent variable vectors.
Split the dataset into test and training set
Apply feature scaling to standardize the independent variables range
end for
for j ← 1 *to* M **do**
while p>=n
Apply LDA to the test and train independent variables and fit it to the model to reduce the dimensionality to p.
end while
Apply the appropriate classifier for both training and test set.
end for
Compute the confusion matrix to test accuracy for a different model where Ω is accuracy.
if Ω model-x > Ω model-y **then**
Model-x
else
Model-y
end if
Construct the flood forecast.

---

5 Results And Discussion

Since we have performed LDA and decomposed the independent data we can visualize the data clusters in two-dimensions. Where LD1 and LD2 are the dimensions derived from performing the LDA. We see that the plane is clustered for different sets of the algorithm. The dependent variable which has three states are mapped onto these clusters and the prediction value is compared with the cluster boundaries. The training was set to consist of 25 percent of the dataset.

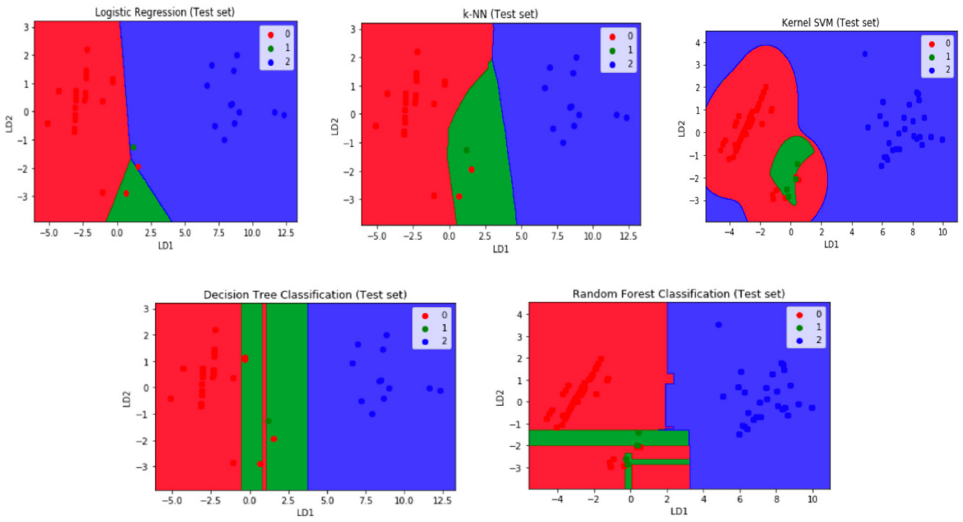

Figure 2. Model comparison on test dataset

Using the confusion matrix True Positives are computed for all the models. Here we observe that Logistic Regression fares the worst among the models for this particular use case. Both k-NN and Random forest classification have very close accuracy scored. Which means they have lower false positive rates. But looking at the visualization of a two-dimensional space we can see that the false positive deviations of the k-NN classifier seem to deviate slightly farther than Random forest classifier. Also, the accuracy obtained with the Random Forest classifier has a slighter better accuracy rate.

Table 2. Accuracy comparison of flood prediction with various classifiers

Methods	Accuracy of classifications
Logistic Regression	0.9562
k-NN	0.9974
Kernel SVM	0.9860
Decision Tree Classification	0.9873
Random Forest Classification	**0.9987**

6 Conclusion

This paper involves the application of multiple machine learning models and uses the model ranking to compute the better model among the several models. This involves a multi-model approach although for the scenario we see that the Random Forest Classification has the higher accuracy. The evaluation of the performance of different models like k-NN, SVR, Decision Tree Classification, Logistic Regression and Random Forest Classifications is very important to choose the right model for the appropriate scenario. The dataset used for this experiment involved weather data for a comparatively shorter time frame.

As the weather is a very complex phenomenon which constitutes of multiple variables and can have other external factors that could affect it. Also, in this particular case, the small shortfall in the accuracy is because of the limited night time sensor data where the current reading from the solar panel shows zero reading. Which the models might interpret as heavy cloud instead of night time. To overcome this UV sensor might be placed which allows determining the day or night cycle, by determining the intensity of sunlight or the lack thereof. During the night the UV levels fall down to negligible levels.

As the data is collected from very few sensor nodes from one city, the diversity in the data is reduced. If the sensor nodes are more diversely scattered and more data is extracted this model can help in better forecasting the Urban flooding.

Thus, by using a hybrid multi model comparison approach when it comes to predicting the data nodes the accuracy of the final model is increased.

ACKNOWLEDGEMENT

The authors would like to place on record their gratitude to authorities of Supercomputer Education and Research Centre, Indian Institute of Science, Bangalore, India for the encouragement and support during the entire course of this work. We also thank the Government of Karnataka for providing us with the necessary permissions to set up the test sensor modules across the city of Bangalore.

REFERNCES

[1] Real-time urban flood forecasting and modelling – a state of the art Justine Henonin, Beniamino Russo, Ole Mark, Philippe Gourbesville Journal of Hydroinformatics Jul 2013, 15 (3) 717-736;

[2] Peter, Lamovec, Mikoš Matjaž, and Oštir Krištof. "Detection of flooded areas using machine learning techniques: Case study of the Ljubljana moor floods in 2010." Disaster Advances 6.7 (2013): 4-11.

[3] Gupta, Anil K., and Sreeja S. Nair. "Urban floods in Bangalore and Chennai: risk management challenges and lessons for sustainable urban ecology." Current Science (2011): 1638-1645.

[4] Ramaswamy, Bhakthavathsalam. "An Intelligent Wireless Modular System for Effective Disaster Management." Transactions on Networks and Communications 4.3 (2016): 22.

[5] Nicholls, N. "A possible method for predicting seasonal tropical cyclone activity in the Australian region." Monthly Weather Review 107.9 (1979): 1221-1224.

[6] Martínez, Aleix M., and Avinash C. Kak. "Pca versus lda." IEEE transactions on pattern analysis and machine intelligence 23.2 (2001): 228-233.

[7] Rao, R. C. (1948). "The utilization of multiple measurements in problems of biological classification". Journal of the Royal Statistical Society, Series B. 10 (2): 159–203. JSTOR 2983775.

[8] Hofmann, Thomas; Scholkopf, Bernhard; Smola, Alexander J. (2008). "Kernel Methods in Machine Learning".

[9] *Altman, N. S. (1992). "An introduction to kernel and nearest-neighbor nonparametric regression". The American Statistician.* ***46*** *(3): 175–185.*

[10] Rokach, Lior; Maimon, O. (2008). Data mining with decision trees: theory and applications. World Scientific Pub Co Inc. ISBN 978-9812771711.

[11] Ho, Tin Kam (1995). *Random Decision Forests* (PDF). Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, 14–16 August 1995. pp. 278–282.