ORIGINAL ARTICLE

INTERNATIONAL JOURNAL OF CONVERGENCE IN HEALTHCARE

IJCIH
Converging Healthcare & Technology

Published by
IJCIH & Pratyaksh Medicare LLP

www.ijcih.com

Comparative Study of Different Machine Learning Techniques for Early Detection of Forest Fires

Paramita Guha[1], Chetna Tyagi[2]

[1]Scientist, CSIR-National Physical Laboratory, Delhi, [2]Assistant Professor, Department of Applied Sciences, The NorthCap University, Gurugram, Haryana, India

Abstract

This paper considers development of an efficient machine learning algorithm for early detection of forest fires. The prominent environmental factors are identified and measured using suitable sensors. The data had been preprocessed and several machine learning algorithms are developed for them. Here, the authors have considered SVM, K-NN and Random Forest algorithms for analysis of the data. To estimate the quality of the algorithms, accuracy, f1 score, precision, recall, confusion matrix etc. had been obtained. Finally, a comparison had been made. Numerical results along with the discussions are also given in this paper.

Keywords: *Classification, SVM, KNN, Random Forest, Precision, confusion matrix, forest fire.*

Introduction

Forests fires are unplanned and uncontrolled fires that occur in natural wooded areas. These fires can occur in both hot and humid conditions and can cause significant damage to the environment, wildlife, and human communities. It can be caused by natural factors such as lightning, volcanic eruptions, or dry weather or it can be started by humans who lit unknowingly or intentionally like campfires, cigarettes, and fireworks. These fires can spread quickly and can be dangerous for fire-fighters and individuals living in nearby areas. It is essential to have proper management plans and strategies in place to prevent and fight forest fires, including early detection and suppression measures, controlled burns, and educating the public on how to prevent forest fires.

These types of fires can be classified into three main categories based on their behaviour *viz.* surface fires, crown fires and ground fires. Surface fires are characterized by flames that burn along the forest floor, fuelled by dry vegetation, such as fallen leaves and branches. Surface fires are often relatively slow-moving and can be controlled with minimal effort. Crown fires occur when flames reach the uppermost branches and foliage of trees, often spreading rapidly and with great intensity. These fires can be difficult to control and pose a significant threat to both wildlife and humans. And lastly, ground fires occur underground, burning in the organic layer of soil or peat. Ground fires are often difficult to detect and can smoulder for weeks or even months, creating a risk of re-ignition. In addition to these categories, forest fires can also be classified based on their cause, such as lightning strikes, human activity, or controlled prescribed burns[1].

These fires can cause significant damages both to the

Corresponding Author:
Paramita Guha
Scientist, CSIR-National Physical Laboratory, Delhi
e-mail: paramita.guha@nplindia.org

forest ecosystem and to human communities neighboring the forest. Here are some major damages occur due to forest fires.

Environmental damages: Forest fires can destroy the natural habitat of animals and plants, altering the entire ecosystem for years to come. The burning of organic material also releases large amounts of carbon dioxide, contributing to climate change.

Economic damages: Forest fires can cause significant economic losses. For example, logging companies may lose millions of dollars' worth in timber or other resources. Additionally, the disruption of transportation and the forcing evacuation of local residents can also result in significant economic impacts. *Health damages:* Forest fires can severely impact public health, especially those living close to the affected area. Smoke could cause respiratory problems or other health issues, meanwhile the wildfire evacuation can lead to mental health problems like PTSD.

Property damages: Properties, houses, or infrastructure located far from the forest, but still in the fire's path can be destroyed, causing a significant financial loss. Also, the extent of the damage varies depending on factors like the location of the fire, the intensity of the blaze, the duration of the fire, and the weather conditions. Hence, inability to predict such fires in well advanced can cause immense losses in properties and environments. Thus an efficient algorithm is hour of need which can predict the fire-like situations based on the environmental parameters. Machine learning algorithms can play important role in this regard. In this paper, authors have collected different environmental parameters like, CO, CO2, smoke, temperature, relative humidity and developed several machine learning models on them so that forest fires can be identified at its initial stage. The paper is organised as follows. In Section II, the problem is discussed. Different machine learning algorithms used are given in Section III, with results are shown in Section IV. Finally the paper is concluded with the subsequent section.

Problem Statement: It has been observed from the literature that the parameters, like, temperature, humidity, carbon dioxide and carbon monoxide, smokes are most significant for the early detection of fires. The parameters during fire situations are given in Table 1. Since, change in methane is very less, hence this parameter is not considered in our experiment for the time being.

Table 1: Environmental parameters during forest fires [1]

	Normal days	Before fire	During fire	After fire
Temperature (°C)	25-30	20-30	60-80	>100
Humidity (%)	50-60	50-60	20-30	20-30
CH4 (ppm)	1.5-2	5-10	5-10	10
CO (ppm)	0.1-0.5	0.5	40-50	20-30
Smoke (ppm)		1-100	>70	100
CO2 (ppm)	100	100	>500	400

Now after collecting the data using suitable sensors, different machine learning algorithms are applied on them. After visualising the data, authors have chosen random forest, k-nearest neighbour (KNN), and Support Vector Machine (SVM) algorithms for analysis of the data. Finally, a comparative study has been made. In the next Section, the machine learning algorithms used are discussed in brief.

Machine Learning Algorithms

(a) **Random Forest:** Random forest is a machine learning algorithm used for both classification and regression tasks. It is an ensemble learning method that combines multiple decision trees to create a more accurate and robust model. Each decision tree in the random forest is trained on a randomly selected subset of features and data points from the training set, which helps to reduce overfitting and improve the model's generalizability[2-4].

(b) During the prediction phase, each decision tree in the forest independently generates its own prediction, and the final output is determined by aggregating the predictions of all the trees in the forest, for example, by taking the majority vote for classification tasks or the mean value for regression tasks. This algorithm has several advantages, including handling missing values and noisy data, being highly scalable, and requiring minimal feature engineering. It is widely used in various applications, such as image and speech recognition, fraud detection, and bioinformatics analysis.

(c) KNN: K-Nearest Neighbors (KNN) is a type of supervised machine learning algorithm that can be used for classification or regression tasks. The algorithm works by finding the k nearest data points to a given input, and using those points to make a prediction about the output of the input. In a classification task, the algorithm would take the most common label among the k nearest neighbors as the predicted label for the input. In a regression task, the algorithm would take the average of the k nearest neighbors as the predicted output for the input[5-8].

KNN is based on the idea that similar inputs are likely to have similar outputs. The algorithm is simple to implement and understands, but it can be computationally expensive, especially for large datasets. It is also sensitive to the number of neighbors chosen (k), and the distance metric used to measure similarity between inputs.

(d) SVM: SVM stands for Support Vector Machine, which is a type of machine learning algorithm used for classification and regression analysis. In SVM, data points are plotted in a high-dimensional space, and the algorithm finds a hyperplane or boundary that separates the data points based on their class labels. The hyperplane is chosen in such a way that the margin between the nearest data points of each class is maximized, leading to better generalization and robustness of the model.

(e) It can handle both linear and non-linear data and can be applied to various types of data analysis problems, such as image classification, text classification, and bioinformatics. SVM is also known for its ability to handle high-dimensional data and is often used in feature selection and dimensionality reduction[9-13].

Results and Discussion

Sensor nodes were placed at strategic locations to collect the environmental parameters at different situations viz., no fire, fire like situation and fire. The dataset had been classified accordingly. It had then been cleansed; noises eliminated and missing values replaced. The pair-plots of the dataset are shown in Fig. 1.

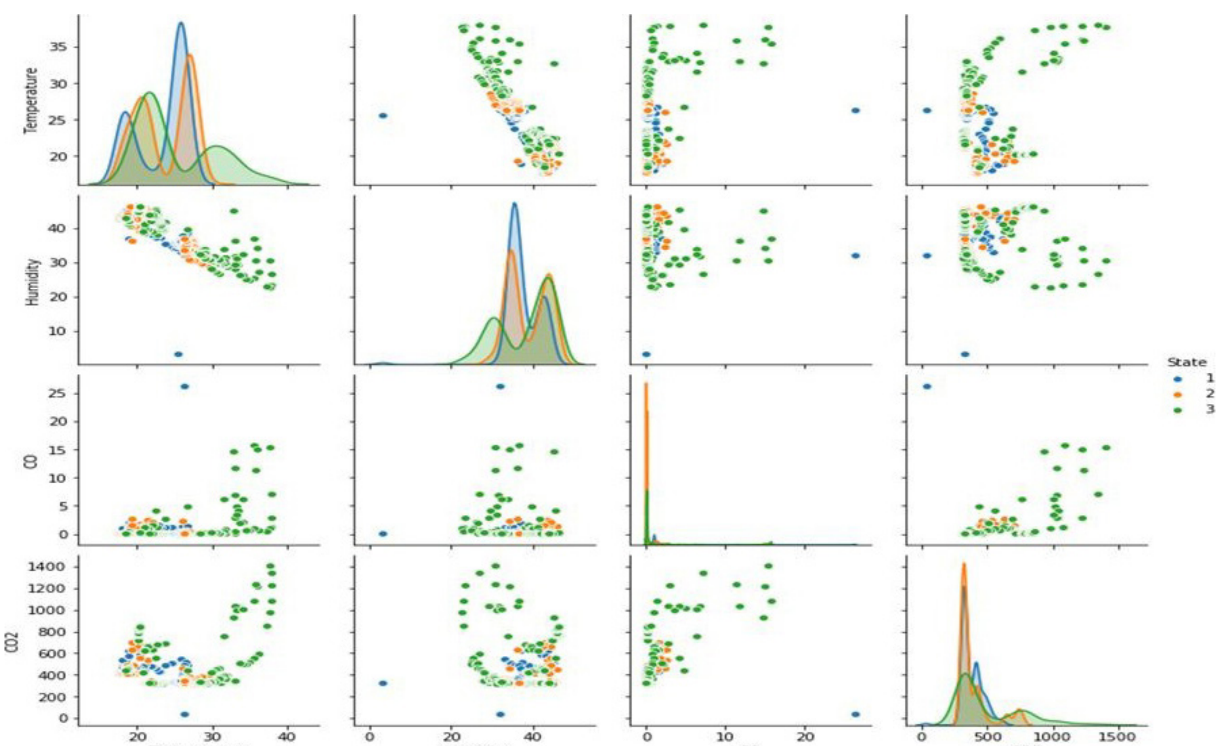

Fig. 1: Pairplots of datasets

Python Codes were written for the machine learning algorithms to implement the problem statement. For the evaluation, the dataset is divided into two parts. 80% of the data was used for development of the training model whereas the rest data is used for testing of the developed model. After implementation of the algorithms, different parameters like accuracy, precision, recall and f1 score were obtained. The confusion matrix and classification report were also generated.

Different kernels like, linear kernel, polynomial kernel, Radial Basis function, sigmoid kernel were used in SVM for the training dataset with cost function as 100. The accuracy, precision, recall, F1 score for weighted average were obtained for the test dataset and are given in Table 2. The confusion matrix for SVM algorithm for the given dataset is shown in Figure 2.

Table 2: Comparison of Linear & Non Linear Kernel SVMs

Kernel Used	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
Linear	61.02	60	59	60
Polynomial	49.17	49	48	45
Radial basis function	45.78	45	46	45
Sigmoid	37.74	36	37	36

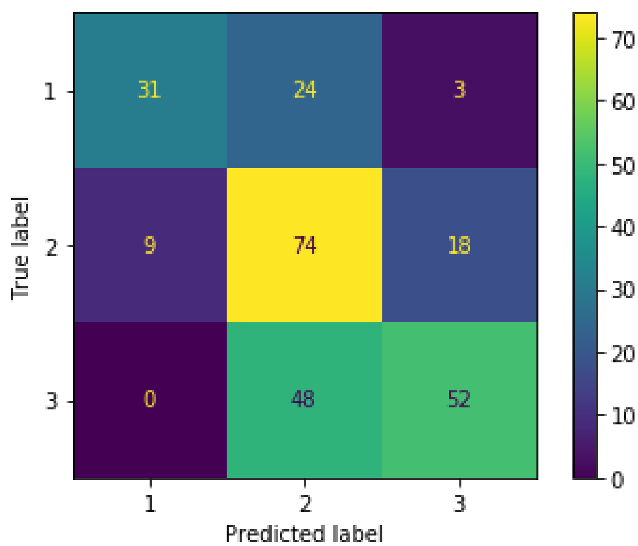

Fig. 2: Confusion Matrix for SVM

For obtaining the K-NN model, again the dataset was opened from CSV and split into training and test datasets. The similarity of the dataset was checked by calculating the distance between the two data instances. An important step in KNN algorithm is to determine the value of *k*. The following points were taken care while choosing the value of *k*.

Smaller values for K can be noisy and have a higher influence on the outputs.

- Larger values of K can have a smoother decision boundary which means lower variance but increased bias. Also, computationally expensive.
- In cases where we are taking a majority vote (e.g. picking the mode in a classification problem) among labels, we usually make K an odd number to have a tiebreaker.
- In general, practice, choosing the value of k is k = sqrt(N) where N is the number of samples in the training dataset. For our model, $N = \sqrt{12930} \cong 113$

The outputs with the K-NN model are given in Figure 3.

```
Testing set Accuracy:88.54961832061069%
1:no_fire, 2:fire_like_situation, 3:fire
[[49  7  3]
 [16 79  5]
 [ 8 15 77]]
              precision    recall  f1-score   support

           1       0.67      0.83      0.74        59
           2       0.78      0.79      0.79       100
           3       0.91      0.77      0.83       100

    accuracy                           0.79       259
   macro avg       0.79      0.80      0.79       259
weighted avg       0.80      0.79      0.79       259
```

Fig. 3: Output console window for KNN

The confusion matrix for the algorithm is given in Figure 4.

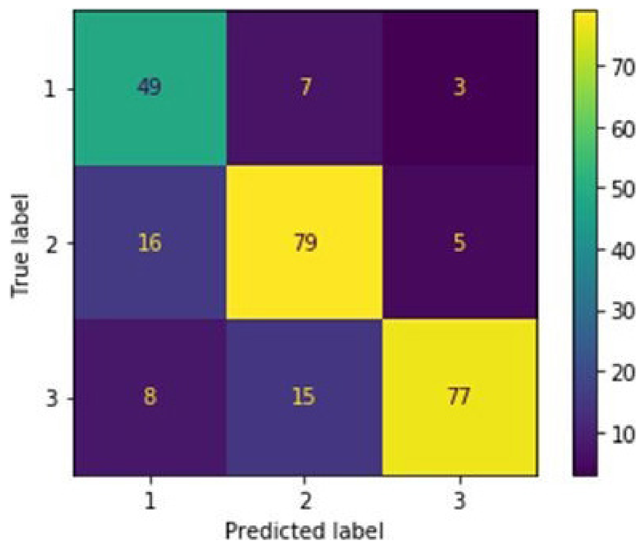

Figure 4: Confusion Matrix for KNN

For implementation of Random Forest algorithm, the following steps were adopted.

The dataset was opened from CSV and split into test/ train datasets

- Bootstrapping had been used for the selection of random samples from the given dataset.
- Decision trees had been constructed for every sample.
- Prediction results were obtained from every decision tree. In this step, voting had been performed for every predicted result.
- At last, the most voted prediction result had been selected as the final prediction result. The outputs are shown in Figures 5 and 6.

```
Random Forest
Train Accuracy= 1.0
Test Accuracy= 0.9922779922779923
[[ 56   0   2]
 [  0 101   0]
 [  0   0 100]]
              precision    recall  f1-score   support

           1       1.00      0.97      0.98        58
           2       1.00      1.00      1.00       101
           3       0.98      1.00      0.99       100

    accuracy                           0.99       259
   macro avg       0.99      0.99      0.99       259
weighted avg       0.99      0.99      0.99       259
```

Fig. 5: Output console window for Random Forest

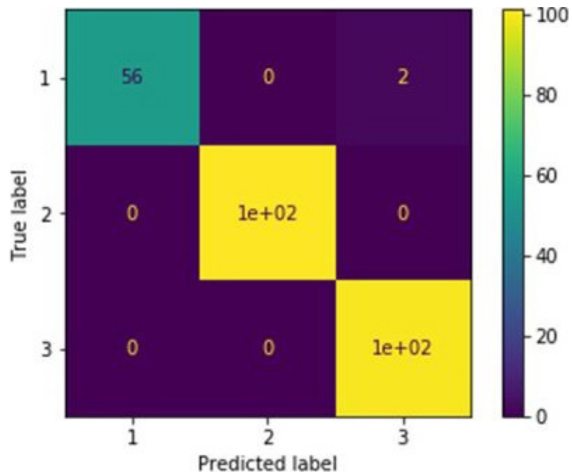

Figure 6: Confusion Matrix for Random Forest

A comparison of all the algorithms is shown below. From the results it can be seen that both K-NN and Random forest algorithms work good compared to SVM algorithms.

	Precision	Recall	F1-score
Fire	0.89	0.86	0.87
Fire like situation	0.85	0.72	0.78
No fire	0.62	0.95	0.75
Accuracy			
Macro avg	0.79	0.84	0.80
Weighted avg	0.83	0.81	0.81

Results for Random Forest

	Precision	Recall	F1-score
Fire	0.78	0.81	0.79
Fire like situation	0.87	0.85	0.86
No fire	0.90	0.90	0.90
Accuracy			
Macro avg	0.85	0.85	0.85
Weighted avg	0.86	0.86	0.86

Results for k-Nearest Neighbour

	Precision	Recall	F1-score
Fire	0.74	0.54	0.62
Fire like situation	0.53	0.75	0.62
No fire	0.71	0.49	0.58
Accuracy			
Macro avg	0.66	0.59	0.61
Weighted avg	0.64	0.61	0.61

Results for SVM

Conclusion

The datasets are obtained from strategically placed sensor nodes inside an experimental field. After primary analysis on the datasets, it is found as a classification problem and suitable algorithms were to be selected. Using the MATLAB classification learner app, SVM, KNN and random forest are found to be efficient. Further, programs for the algorithms are written and evaluated in python language. Various performance parameters are studied. The accuracy and f1 score for random forest algorithm is found to be better than KNN algorithm. Also, error rate and root mean square error as a function of the number of neighbors are found to be lesser for random forest algorithm. However, both K-NN and Random Forest algorithms are more efficient than SVM.

Conflict of Interest: There is no conflict of interest among the authors on this article.

Source of Funding: This work was funded by DST under the project DST/INT/SL/P-19/2016

Ethical Clearance: Not Required

References

1. Chowdary V, Gupta M, Singh R. A review on forest fire detection techniques: A decadal Perspective". Int J Eng Technol, vol. 7. 2018;3(12):1312-6
2. Ben-Haim Y, Tov E, Breiman L. A streaming parallel decision tree algorithm. Mach Learn. 2001;45(1):5-32.doi:10.1023/A:1010933404324.
3. Jijo B, Abdulazeez A. Classification based on decision tree algorithm for machine learning. J Appl Sci Technol Trends. 2021;02(01):20-8.
4. Ho T. The random subspace method for constructing decision forests. IEEE Trans Pattern Anal Mach Intell. 1998;20(8):832-44. doi: 10.1109/34.709601.
5. Wettschereck D, Thomas D G. Locally adaptive nearest neighbour algorithms. Adv Neural Inf Process Syst. 1994:184-6.
6. Han EH, Karypis G, Kumar V. 2001. Text categorization using weight adjusted k-nearest neighbour classification. Advances in knowledge discovery and data mining. PAKDD 2001, lecture notes in computer science, vol 2035. Berlin, Heidelberg: Springer.
7. Jiang S, Pang G, Wu M, Kuang L. An improved K-nearest-neighbour algorithm for text categorization. Expert Syst Appl. 2012;39(1):1503-9. doi: 10.1016/j.eswa.2011.08.040.
8. Sun S, Huang R," An adaptive k-nearest neighbour algorithm" Seventh International Conference on Fuzzy Systems and Knowledge Discovery. Vol. 2010; *2010*.
9. Li J, Allinson N, Tao D, Li X. Multitraining support vector machine for image retrieval. IEEE Trans Image Process. 2006;15(11):3597-601. doi: 10.1109/tip.2006.881938, PMID 17076417.
10. Pontil M, Verri A. Support vector machines for 3D object recognition. IEEE Trans Pattern Anal Machine Intell. 1998;20(6):637-46. doi: 10.1109/34.683777.
11. Burbidge R, Trotter M, Buxton B, Holden S. Drug design by machine learning: support vector machines for pharmaceutical data analysis. Comput Chem. 2001;26(1):5-14. doi: 10.1016/s0097-8485(01)00094-8, PMID 11765851.
12. Valentini G. Gene expression data analysis of human lymphoma using support vector machines and output coding ensembles. Artif Intell Med. 2002;26(3):281-304. doi: 10.1016/s0933-3657(02)00077-5, PMID 12446082.
13. Shutao L, Kwok JT, Zhua H, Wang Y. Texture classication using the support vector machines. Pattern Recognit. 2003;36:2883-93.