International Journal of Electrical and Computer Engineering (IJECE)
Vol. 11, No. 3, June2021, pp. 2535~2547
ISSN: 2088-8708, DOI: 10.11591/ijece.v11i3.pp2535-2547

A new procedure for misbehavior detection in vehicular ad-hoc networks using machine learning

Abhilash Sonker, R. K. Gupta
Department of Computer Science and Engineering and Information Technology,
Madhav Institute of Technology and Science, Gwalior, India

Article Info

Article history:

Received Sep 4, 2020
Revised Oct 6, 2020
Accepted Dec 5, 2020

Keywords:

Binary classification
Machine learning
Misbehavior detection
Multiple classifications
VANETs

ABSTRACT

Misbehavior detection in vehicular ad hoc networks (VANETs) is performed to improve the traffic safety and driving accuracy. All the nodes in the VANETs communicate to each other through message logs. Malicious nodes in the VANETs can cause inevitable situation by sending message logs with tampered values. In this work, various machine learning algorithms are used to detect the primarily five types of attacks namely, constant attack, constant offset attack, random attack, random offset attack, and eventual attack. Firstly, each attack is detected by different machine learning algorithms using binary classification. Then, the new procedure is created to do the multi classification of the attacks on best chosen algorithm from different machine learning techniques. The highest accuracy in case of binary classification is obtained with Naïve Bayes (100%), decision tree (100%), and random forest (100%) in type1 attack, decision tree (100%) in type2 attack, and random forest (98.03%, 95.56%, and 95.55%) in Type4, Type8 and Type16 attack respectively. In case of new procedure for multi-classification, the highest accuracy is obtained with random forest (97.62%) technique. For this work, VeReMi dataset (a public repository for the malicious node detection in VANETs) is used.

Corresponding Author:

Abhilash Sonker
Department of Computer Science and Engineering and Information Technology
Madhav Institute of Technology and Science
GolekaMandir, Gwalior (474005), MP, India
Email: abhilashsonkerit@mitsgwalior.in

1. INTRODUCTION

Vehicular ad hoc networks (VANETs) are similar to mobile ad hoc networks (MANETs) [1]. VANETs are produced by applying the principles of MANETs. VANETs have nodes which communicate to each other through message logs and are short lived [2]. All nodes share same radio channel and exchange data with other nodes [3]. Message logs consist of several features like sending time, sending Id, message Id, position, noise in the position, speed, noise in the speed, etc. Small packets are repeatedly exchanged with the other nodes in neighborhood to maximize safety in automobile driving [4]. Traditional wired network gives protection by different methods like gateways, firewalls, etc. However, wireless networks are liable to security attacks aiming the whole network from different directions. Because of different misbehaviors like spamming, bluffing, faking of identities will generate malignant nodes which can lead to transfer incorrect or inaccurate messages to the neighborhood nodes this will decrease the performance of VANET as well as road safety and increased road accidents can be seen. Looking forward safety of a passenger can be enhanced by means of inter-vehicle communication [5]. For example, if any road accident occurs, with the help of

VANET communication safety alert packets are transferred when a node notices a censorious event, this will make other vehicles alert moving towards that site; with this road accidents can be minimized [6]. In this way, duty of honest nodes is to forward each accepted risk-free packets to the nodes in its transmission range.

An application of VANET support actual time communication and mainly deals with critical information related to life [7]. So as to achieve correctness and effectiveness, it should stick to security demand that is honesty, non-repudiation, privacy, confidentiality and authentication to shield against the attacker and malicious nodes [8]. To come up with preventive measures, observation of such malicious nodes and unusual activities in the network is very much important. At the end of the day stunning growth of road traffic in worldwide; it becomes very crucial to use current technologies to make safer and easier driving for the driver.

In this paper, there are five types of attacks such as constant attack (type1), constant offset attack (type2), random attack (type4), random offset attack (type8) and eventual attack (type16). the constant attacker transfers fixed, pre-configured position; the constant offset attacker transfers fixed, pre-configured offset added to their actual position; the random attacker sends a random position; the random offset attacker transfers a random position in a pre-configured triangle in a vehicle, the eventual attacker behaves normal for the sometime repeatedly.

The current work is to present a new procedure for misbehavior detection in VANETs using machine learning techniques. In this paper machine learning is going to help in classification of message logs send from a node to be honest or malicious. For the classification various features are extracted from the nodes. With the help of nearby nodes these features are calculated. After calculation observations are inter-changed by the observer nodes to the other nodes in its neighborhood. In this paper two types of machine learning classification techniques are used that is binary classification and multi-classification. The accuracy of machine learning model firstly depends on the algorithm that is used to generate the classifier and secondly the features that are used to represent the instances. Different inducers and features give different performance for each classifier [3]. To overcome this new system is created so that best algorithm is automatically chosen according to the dataset and when new message log is sent from the node it is detected that message log has any type of attack or not. If it is founded that message log is malicious then the node from which this message log is transferred is also malicious and hence detected.

The approaches which are done to detect the misbehavior in VANETs are mostly simulation based. In recent years the use of ad-hoc network rises tremendously [9]. The automating the systems to detect the misbehavior in VANETs will give an aid to detect them on live environment. Here we are discussing some of the works which have been done to enhance the node detection system in VANETs.

Grover *et al.* used Naive Bayes, IBK, AdaBoost1, J-48 RF to predict misbehavior in VANETs. But random forest and J-48 gave the best results. Dataset was consisting of 3101 legitimate and 1427 malicioussamples. Results were based on metrics with high values of TPR (0.93), TNR (0.99) and small values of FPR (0.005) and FNR (0.06) [1]. Khana *et al.* (2014) presented network simulation based study a topic on detection of malicious nodes (DMN) in vehicular ad-hoc networks. They proposed a novel algorithm called DMN (detection of malicious nodes in VANETs) to detect malicious nodes [2].

Again, in 2012, Grover *et al.* presented a concept of misbehavior detection based on ensemble learning in VANET. Algorithms used were Naive Bayes, IBK, AdaBoost1, J-48, RF and ensemble based learning. Ensemble based learning gave the highest accuracy TPR (0.95), FPR (0.01) and TNR (0.99), FNR (0.03) [10]. Muthukumar and Karthick presented a topic on identifying the misbehavior nodes using trust management in VANETs. In this article they have introduce some misbehavior prevention researches in location privacy-enhanced VANETs. In the future, they have intended to improve the detection rate of the proposed system and to evaluate the performance of the proposed scheme with different vehicle densities and average velocities [11]. Barnwal and Ghosh present a survey on detection of misbehaving nodes in vehicular ad-hoc network and conclude to adopt hybrid based techniques for misbehavior detection [12].

Sedjelmaci *et al.* presented a topic on predict and prevent from misbehaving intruders in heterogeneous vehicular networks to prevent the occurrence of the most dangerous attacks that target HetVNet. They have analyzed the performances and demonstrated the efficiency of the proposed scheme using NS-3, which showed that it exhibits a high accuracy prediction rate, low detection time and a low communication overhead [13]. Mohammadi *et al*. conducted a survey on misbehavior node detection in vehicular ad-hoc networks. Compared to SVM-based, dempster shafer-based, and averaging-based detection techniques. SVM classifier gives the highest accuracy [14]. Tiwari and Gupta conducted a survey on security enhancement of misbehavior nodes in vehicular ad-hoc networks using hash function; algorithms used were J-48, RF, IBK, Naïve Bayes and AdaBoost1. But J-48 and RF gave thebest results [15].

It is found that only few works has been done regarding the malicious node detection in VANETs using machine learning. This study is going to give a research support as well for the future aspirants who

want to study in this field. This paper is written with the aim of providing a procedure for the selection of best algorithm from different algorithms for misbehavior detection in VANETs.

2. RESEARCH METHOD

Vehicular reference misbehavior (VeReMi) dataset is a dataset for evaluation of the misbehavior detection in the VANETs. This dataset is composed of two types of files ground truth file and the message logs generated from the simulation environment. It is a part of recently published paper [16]. It is simulated generated using LuST and VEINS. It primarily discussed five attacks namely, constant attack (type1 attack), constant offset attack (type2 attack), random attack (type4 attack), random offset attack (type8 attack), and eventual attack (type16 attack). Its primary purpose is to serve as a baseline to assess how misbehavior detection mechanisms operate on a city scale. The dataset contains the five different files of the different types of attacks having 960, 1056, 4438, 21638, 20483 with initial instances in type1, type2, type4, type8 and type16 attack respectively. The combined dataset have 48,575 instances for the multi classification. The memory size of total dataset is approx 5.3 MB.

The research is carried out in two different phases; the first phase is for the analysis of the algorithms on different attacks and second phase is to design a new procedure for the detection of attacks using the different machine learning classification algorithms like Naïve Bayes, K-nearest neighbor (KNN), stochastic gradient descent (SGD) classifier, decision tree (DT) and random forest. Each algorithm is applied and accuracy is evaluated in the first phase for the individual attack. Let us understand the each algorithm one by one.

2.1. Naïve Bayes

The advancement in the Bayesian theory gets the evolution of Naïve Bayes algorithm. The Naïve Bayes is a supervised machine learning algorithm based on the Bayes Theorem [17]. The Bayes theorem for the likelihood is given as (1):

$$P\left(Y/X\right) = \frac{P\left(X/Y\right) * P(Y)}{P(X)} \quad (1)$$

Since in (1) the $P(X)$ is constant and add extra calculation in the computation, hence it is being removed from the formula, and given as (2):

$$P\left(Y/X_i\right) = \sum_i P\left(X_i/Y\right) * P(Y) \quad (2)$$

In (2) gives the result of the Naïve Bayes classifier.

2.2. K-nearest neighbor

K-nearest neighbor is called the instance based learner as it stores the instances for classification. The K-Nearest Neighbor classifier works on the principle of majority voting. In this algorithm K is the number of nearest neighbors to be considered. The distance of each element is calculated from the query point and identifies the class of each neighbor. Then based on the majority voting the query point is classified. This algorithm is also known as lazy learning algorithm because after training the model it waits for the query point [18]. The formula used in the calculation of the distance of the nearest neighbors is Euclidean distance:

$$\sqrt{\sum(x_m - y_m)^2} \quad (3)$$

In this work, (3) that is Euclidean distance is used for the calculation of the distance.

2.3. Stochastic gradient descent

Stochastic gradient descent is not the actual algorithm of the supervised machine learning. It is an optimization technique. This technique is efficient for the solving linear problems with support vector machines and logistic regression. The work presented is convex optimization of support vector machine. It is widely used because of efficiency and ease [19]. The stochastic gradient descent in contrast a perform a parameter update for each (x, y) is given by (4):

$$\theta^{new} = \theta^{old} - \alpha \nabla_\theta J(\theta; x; y) \quad (4)$$

SGD removes the recomputation of gradients for similar examples and hence it is faster and can be used to learn online [20].

2.4. Decision tree

The decision trees are ways to find the conclusion based on the set of rules drawn from the tree. A decision tree is consists of the two nodes: i) Decision node and ii) Leaf node. A decision node tells about which attribute have to be selected and leaf node tells about the class. Decision trees use the up down approach to give the results [21]. The first node of the decision trees, a decision node, called as root node [22]. Each node of the decision tree is selected on the basis of information gain methods:

2.4.1. Information gain method

The two important formulas that are used in this method are: i) Entropy calculation and ii) Information gain, for calculating the entropy of the sample data

$$E(s) = \sum -p_i \log_2 p_i$$

After calculating the entropy, the information gain is calculated for each attribute to get decide the decision node.

$$Gain(S, A) = Entropy(S) - \sum_{v \in values(A)} |S_v|/|S| Entropy(S_v)$$

2.5. Random forest classifier

Random forest is an ensemble technique. It is called as bootstrapping and aggregation is the result based on majority vote of base models on the test data [23]. Random forest is a bagging technique which feed the data to the base models by row sampling with replacement and predicting the classes. Usually, decision tree is used as its base model. Random forest applies both feature sampling and row sampling with replacement. The Figure 1 given is an example showing the random forest classification.

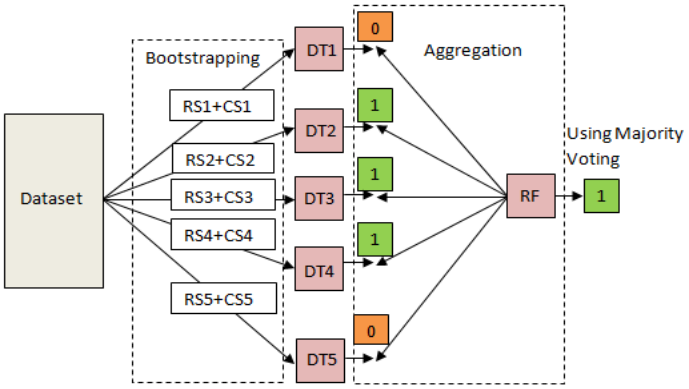

Figure 1. Random forest classification

Suppose training dataset which is being classified into 0 or 1 that is binary classification is given to different decision tree models with the feature sampling and row sampling with replacement then the results by the decision trees are given as shown in the Figure 1. Now when a test dataset is passed then the results of the decision trees aggregates using majority voting method to predict the final class [24]. A decision tree alone when classify a dataset it has low bias and high variance when the tree is grown to the maximum depth. To reduce the variance feature sampling and row sampling is used with different decision tree models.

2.6. New procedure design

After completing the first phase, a new procedure is designed for the detection of attacks on message logs send from any node as shown in Figure 2. This new procedure is including the following steps:

- Selecting the ground truth data from VeReMi dataset
- Loading the dataset to the environment

- Pre-processing of the ground truth data
- Selection of the best features fitting the model
- Fitting the models
- Calculation of accuracy
- Saving accuracies
- Model with highest accuracy
- Passing new message log to the model
- Prediction of attacks

In the pre-processing of the ground truth data, the unnecessary columns are removed and data is balanced. After balancing the data the feature selection is performed to select the important features. After selecting the important features the data is split into training and testing set and model is trained and accuracy is calculated. The accuracy is saved for all the models and best model is selected based on the highest accuracy. After getting the best accuracy model, whenever a new message log comes with suitable features it is going to be detected that which type of attack it has. Finding the attack in the message log it is also detected that node from which new message log is had been sent is also malicious. Hence malicious node is detected.

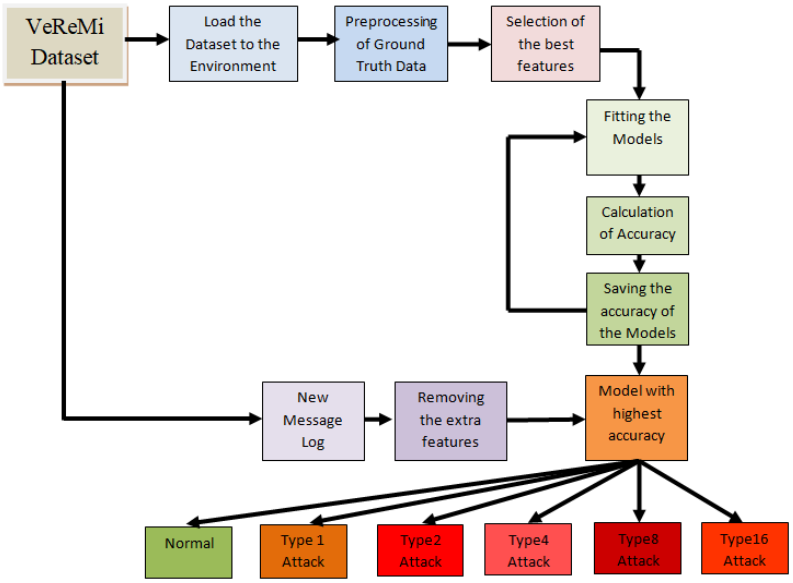

Figure 2. New procedure for misbehavior detection in VANETs

3. RESULTS AND DISCUSSION

The VeReMi dataset contains several folders of different versions and in this work single file is selected from the individual attacks and is analyzed on the different five algorithms namely, Naïve Bayes, KNN, stochastic gradient descent, decision tree and random forest. Using the confusion matrix the accuracies are calculated. The procedure for the checking the individual attacks is same for every algorithm. The accuracy is calculated using the (5) from the confusion matrix [25].

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+FN)+(FP+TN)} \quad (5)$$

Where TP = True Positive
TN = True Negative
FP = False Positive
FN = False Negative

The classification report is also given for every algorithm containing the precision, recall, F1 score and support. The precision gives the ratio of correctly predicted positive operations to the total predicted positive observations. The recall is also called as sensitivity and gives us idea of the true positive observations to the total actual positive observations. F1 score gives the weighted average of precision and recall.

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1 score} = \frac{2 * (Precision * Recall)}{(Precision + Recall)}$$

Let's discuss the result of each attack one by one.

3.1. Type1 attack (constant attack)

The dataset of constant attack or type 1 attack was taken and preprocessing and feature selection is done to create the model. Then models are evaluated and their accuracies are calculated:

3.1.1. Naïve Bayes

The confusion matrix and ROC-curve drawn corresponding to the Naïve Bayes algorithm is shown in the Figure 3.

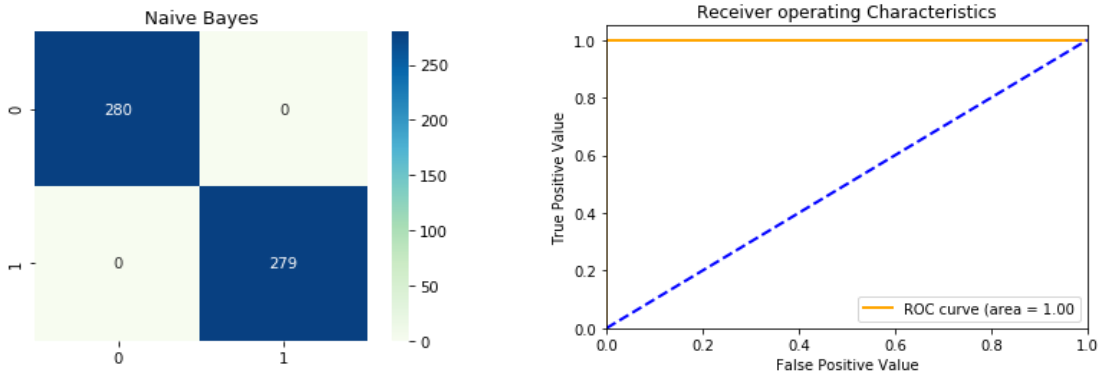

Figure 3. Confusion matrix and ROC-curve of Naïve Bayes for constant attack

The accuracy is calculated using (5) as

$$\text{Accuracy} = \frac{280 + 279}{(280 + 0) + (279 + 0)}$$

$$\text{Accuracy} = \frac{559}{559} = 0.100$$

Hence the accuracy is 100.00%. The classification report is shown in Figure 4. Similarly, other algorithms KNN, SGD, decision tree and random forest are evaluated with accuracies of 99.10% and 97.60%, 100%, and 100% respectively.

Naive Bayes

	precision	recall	f1-score	support
0	1.00	1.00	1.00	280
1	1.00	1.00	1.00	279
accuracy			1.00	559
macro avg	1.00	1.00	1.00	559
weighted avg	1.00	1.00	1.00	559

Figure 4. Classification reports of Naïve Bayes for constant attack

3.2. Type2 attack (constant offset attack)

The constant offset attack is a type 2 attack and its data during the evaluation is divided into the 70% for training and 30% for testing. The algorithms used for the calculation of results discussed.

3.2.1. Decision tree

The confusion matrix and ROC-curve drawn corresponding to the DT algorithm is shown in the Figure 5.

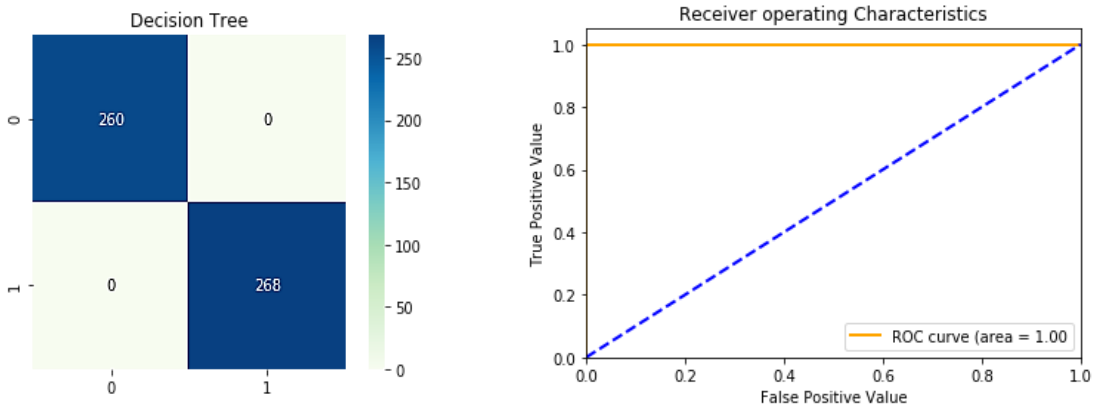

Figure 5. Confusion matrix and ROC-curve of DT for constant offset attack

The accuracy is calculated using (5) as

$$\text{Accuracy} = \frac{260 + 268}{(260 + 0) + (0 + 268)}$$

$$Accuracy = \frac{528}{528} = 0.10000$$

Hence the accuracy is 100.00%. The classification report is shown in Figure 6. Similarly, other algorithms such as Naïve Bayes, KNN, SGD and random forest are evaluated with 77.84%, 95.64%, 76.32% and 99.24% respectively.

Decision Tree

	precision	recall	f1-score	support
0	1.00	1.00	1.00	260
2	1.00	1.00	1.00	268
accuracy			1.00	528
macro avg	1.00	1.00	1.00	528
weighted avg	1.00	1.00	1.00	528

Figure 6. Classification report of DT for constant offset attack

3.3. Type4 attack (random attack)

The random attack is classified with splitting the data into train and test sets. The attack is identified with the help of following algorithms:

3.3.1. Random forest

The confusion matrix and ROC-curve drawn corresponding to the random forest algorithm is shown in the Figure 7.

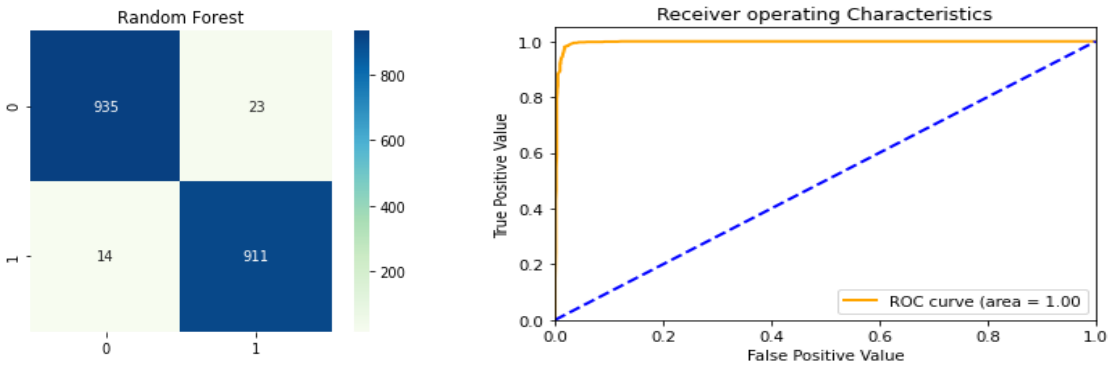

Figure 7. Confusion matrix and ROC-curve of random forest for random attack

The accuracy is calculated using (5) as

$$\text{Accuracy} = \frac{935 + 911}{(939 + 23) + (14 + 822)}$$

$$Accuracy = \frac{1846}{1883} = 0.9803$$

Hence the accuracy is 98.03%.The classification report is shown in Figure 8. Similarly, other algorithms such as Naïve Bayes, KNN, SGD and DT are evaluated with 62.08%, 86.61%, 52.20%, and 96.70% respectively.

Random Forest

	precision	recall	f1-score	support
0	0.99	0.98	0.98	958
4	0.98	0.98	0.98	925
accuracy			0.98	1883
macro avg	0.98	0.98	0.98	1883
weighted avg	0.98	0.98	0.98	1883

Figure 8. Classification report of random for random attack

3.4. Type8 attack (random offset attack)

Type8 attack or random offset attack is taken into consider by splitting the data into 70% training set and 30% testing set. The algorithms show the different results on the same dataset due to their learning function.

3.4.1. Random forest

The confusion matrix and ROC-curve drawn corresponding to the random forest algorithm is shown in the Figure 9. The accuracy is calculated using (5) as

$$\text{Accuracy} = \frac{3941 + 3894}{(3941 + 154) + (210 + 3894)}$$

$$Accuracy = \frac{7835}{8199} = 0.9556$$

Hence the accuracy is 95.56%. The classification report is shown in Figure 10. Similarly, other algorithms such as Naïve Bayes, KNN, SGD and DT are evaluated with 58.83%, 82.58%, 49.03%, and 95.40% respectively.

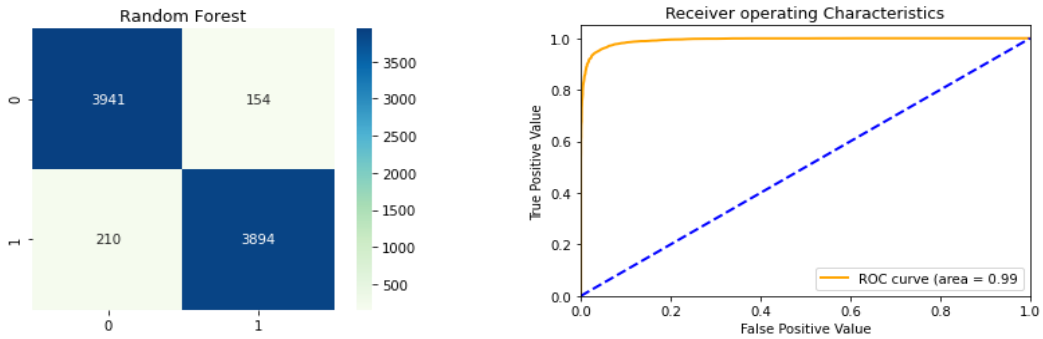

Figure 9. Confusion matrix and ROC-curve of random forest for random offset attack

Random Forest

	precision	recall	f1-score	support
0	0.95	0.96	0.96	4095
8	0.96	0.95	0.96	4104
accuracy			0.96	8199
macro avg	0.96	0.96	0.96	8199
weighted avg	0.96	0.96	0.96	8199

Figure 10. Classification report of random forest for random offset attack

3.5. Type16 attack (eventual attack)

The eventual attack is detected by creating a model using the data for the detection of the eventual attack. Let's calculate the accuracies of each algorithm used to create the models.

3.5.1. Random forest

The confusion matrix and ROC-curve drawn corresponding to the random forest algorithm is shown in the Figure 11.

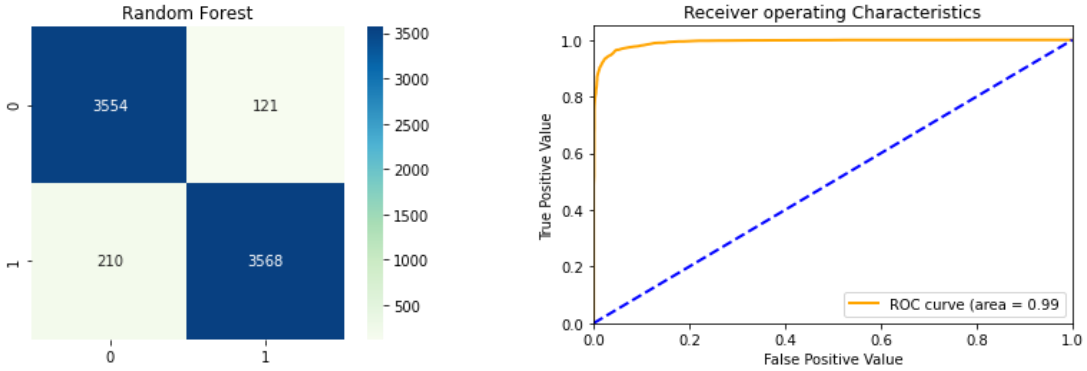

Figure 11. Confusion matrix and ROC-curve of random forest for eventual attack

The accuracy is calculated using (5) as

$$\text{Accuracy} = \frac{3554 + 3568}{(3554 + 121) + (210 + 3568)}$$

$$Accuracy = \frac{7122}{7453} = 0.9555$$

Hence the accuracy is 95.55%. The classification report is shown in Figure 12.

Random Forest

	precision	recall	f1-score	support
0	0.94	0.97	0.96	3675
16	0.97	0.94	0.96	3778
accuracy			0.96	7453
macro avg	0.96	0.96	0.96	7453
weighted avg	0.96	0.96	0.96	7453

Figure 12. Classification report of random forest for eventual attack

Similarly, other algorithms such as Naïve Bayes, KNN, SGD and DT are evaluated with 59.33%, 79.55%, 51.16%, and 94.68% respectively. The results of individual attacks on different algorithms are given in Table 1. For the analysis of the new procedure we have combined all the dataset of the type 1, 2, 4, 8, 16 attacks and make a function to select the best algorithm for the dataset and then choose the best selected algorithm to predict the result for the new dataset.

The discussed algorithms used for the individual attack detection is used in the new procedure. The accuracies of each algorithm are also calculated on the combined dataset. On the analysis of the algorithms and calculating the results it is found that the random forest is giving the highest accuracy in this dataset with 97.62% while the other algorithms like Naïve Bayes, KNN, SGD and decision tree is showing the accuracy of 70.38%, 88.66%, 67.78%, and 95.64% respectively. This whole procedure can be used for the any dataset by changing the feature names and data used for the algorithm to predict the attack. Let's discuss the accuracy of the random forest obtained on the new procedure with highest accuracy. The confusion matrix drawn for the random forest is shown in Figure 13.

Table 1. Results of attacks on different algorithms

Attacks↓ Algorithms→	Naïve Bayes	KNN	SGD	DT	Random Forest
Type1	**100.00%**	99.10%	97.60%	**100.00%**	**100.00%**
Type2	77.84%	95.64%	76.32%	**100.00%**	99.24%
Type4	62.08%	86.61%	52.20%	96.70%	**98.03%**
Type8	58.83%	82.58%	49.03%	95.40%	**95.56%**
Type16	59.33%	79.55%	51.16%	94.68%	**95.55%**

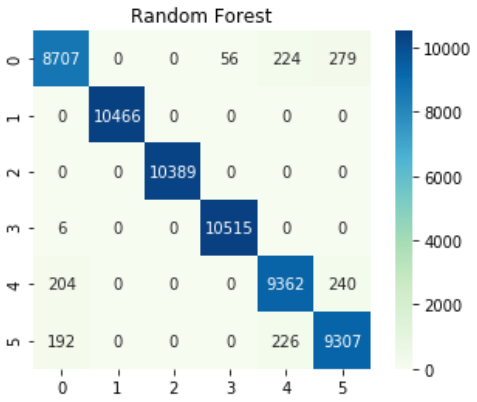

Figure 13. Confusion matrix of random forest for new procedure

$$Accuracy = \frac{8707 + 10466 + 10389 + 10515 + 9362 + 9307}{9266 + 10466 + 10389 + 10521 + 9806 + 9725}$$

$$Accuracy = \frac{58746}{60173} = 0.9762$$

Hence accuracy calculated is 97.62%.

The ROC-curve is drawn between true positive value and false positive value, hence can be drawn for the binary problems by definition. So, it is tedious to draw the curve for multi class classification. However, for multi label classification it is possible to do so. The classification report of the random forest algorithm for new procedure is shown in Figure 14.

Random Forest

	precision	recall	f1-score	support
0	0.96	0.94	0.95	9266
1	1.00	1.00	1.00	10466
2	1.00	1.00	1.00	10389
4	0.99	1.00	1.00	10521
8	0.95	0.95	0.95	9806
16	0.95	0.96	0.95	9725
accuracy			0.98	60173
macro avg	0.98	0.98	0.98	60173
weighted avg	0.98	0.98	0.98	60173

Figure 14. Classification report for the random forest for the new procedure

The machine learning algorithms varies their results on the basis of features, dataset and algorithms used. According to no free lunch theorem [26], "There is no one algorithm that best fit for every problem". Gahleb *et al.* used NGSIM dataset to study the misbehavior in VANETs using artificial neural networks (feed forward neural network and back propagation). They have used the binary classification to detect the misbehavior in every vehicle separately. The features used in this work are overlapped areas, interval to loss received information, average prediction error, distance to the sender, average vehicle occurring distance and vehicle uncertainty. The total accuracy achieved is 99.74% [27]. A comparison of results with proposed work is shown in Table 2.

Bidgoli *et al.* used KDDCUP99 dataset for the intrusion detection using decision tree algorithm on reduced feature space. A study on 41 features and 24 attack types are done of DoS (denial of service), remote to user (R2L), user to root (U2R), and probing class [28]. The comparison with the proposed work is shown in Table 3.

Table 2. Comparison with previous work

Research	Accuracy
Faud A. Gahleb *et al.*	99.74% (ANN)
Proposed Work	100% (DT, Random Forest and Naïve Bayes)

Table 3. Comparison with previous work

Research	Accuracy
Behrouz Minaei Bidgol *et al.*	98.5% (Average for DT in Normal case)
Proposed Work	97.6% (Random Forest)

The comparison of two algorithms should be done on the same dataset in the same environment only then it can be said that the algorithm is best in that scenario with corresponding dataset. In both the above papers the results are definitely little bit varying because of the dataset chosen for the experiment. Hence all the results obtained are correct and best in their own scenario.

4. CONCLUSION

VANETs has gained a lot of attention as it has greatly leaded in the road safety and driving conditions. The misbehavior in the VANETs can be detected to find node to be malicious or not. In this paper, the five attacks are detected by the five different algorithms and accuracy is calculated for each algorithm separately. Although a new procedure is formed for the multiple detection of the attacks using the best algorithm that is possible on the combined dataset. This new procedure can also be work as a general concept or mechanism for the malicious node detection. This approach is suitable for the detection of misbehavior in VANETs by choosing the best algorithm. This algorithm reduces the effort of writing the

codes for the different algorithms separately and doing analysis of each algorithm for choosing the best one. Naïve Bayes with 100% accuracy, decision tree with 100% accuracy and random forest with 100% is obtained in type1 attack. Decision tree with 100% accuracy in type2 attack is obtained. Random forest with 98.03% accuracy in type4 attack, random forest with 95.56% in type8 attack, and random forest with 95.55% is obtained. The new procedure selects the best algorithm as random forest with 97.62%. Hence the new procedure is achieved for getting the best algorithm for the detection of misbehavior in VANETs. The advancement in this paper can be done with the application of the hybrid machine learning techniques. The implementation of the different scenarios and different attacks can also be considered in future for the detection of misbehavior.

REFERENCES

[1] Jyoti Grover, et al., "Machine Learning Approach for Multiple Misbehavior Detection in VANET," *International conference on advances in computing and communications,* vol. 192, 2011, pp. 644-653.

[2] Sherali Zeadally, et al., "Vehicular ad hoc networks (VANETS): Status, results, and challenges," *Telecommunication Systems,* vol. 50, no. 4, pp. 1-25, Aug. 2010

[3] D. Bisen and S. Sharma, "Fuzzy based Detection of Malicious Activity for Security Assessment of MANET," *National Academy of Science Letter,* vol. 41, no. 1, pp. 23-28, 2018

[4] Jyoti Grover, et al., "Attack models and infrastructure supported detection mechanisms for position forging attacks in vehicular ad hoc networks," *CSI Transactions on ICT,* vol. 1, pp. 261-279, 2013.

[5] M. Sameer Sheikh, et al., "Security and Privacy in Vehicular AdHoc Network and Vehicle Cloud Computing: A Survey," *Wireless Communications and Mobile Computing,* vol. 2020, pp. 1-25, 2020

[6] S. Patil and L. Ragha, "Rsadp-Road Safety Accident Detection and Prevention in Vehicular Adhoc Networks," *International Journal of Innovative Technology and Exploring Engineering (IJITEE),* vol. 9, no. 2, pp. 3325-3330, Dec. 2019.

[7] M. Abhijeet Deshmukh and D. Dinesh, "Challenges in Vehicular Adhoc Networks (VANETs)," *International Journal of Engineering Technology, Management and Applied Science,* vol. 2, no. 7, 2014

[8] Uzma Khan, et al., "Detection of Malicious Node (DMN) in Vehicular Adhoc Networks," *International Conference on Information and Communication Technologies (ICICT 2014),* vol. 46, 2015, pp. 965-972.

[9] E. Deshmukh Bisen, et al., "Fuzzy based malicious Detection Approach for Underwater Ad-hoc Wireless Network (UANET)," *Journal of Xiadian University,* vol. 13, no. 10, pp. 1-8, 2019.

[10] Jyoti Grover, et al., "Misbehavior Detection based on Ensemble Learning in VANET," *International Conference on Advanced Computing, Networking and Security,* vol. 7135, 2012, pp. 602-611.

[11] Muthukumar S. and Karthick Selvan R., "Identifying the Misbehavior Nodes Using Trust Management in VANETs," *International Journal of Advanced Research in Computer Science & technology (IJARCT 2014),* vol. 2, no. 1, pp. 271-276, 2014.

[12] Rajesh P. Barnwal and Soumya K. Ghosh, "Detection of Misbehaving Nodes in Vehicular Ad-hoc Network," *book: Security for Multihop Wireless Networks, Chapter: Chapter 6, Publisher: CRC Press,* pp. 139-170, 2014.

[13] Hichem Sedjelmaci et al., "Predict and Prevent From Misbehaving Intruders in Heterogeneous Vehicular Networks," *Vehicular Communications, Elsevier,* vol. 10, pp. 74-83, 2017

[14] Zahra Soltani Mohammadi, et al., "Misbehavior Node Detection in Vehicular ad-hoc Networks: A survey, With Special Emphasis on Multihop Broadcast Protocols," *Researcher,* vol. 9, no. 1, pp. 41-46, 2017.

[15] Priyanka Tiwari and Rahul Gupta, "Security Enhancement of Misbehavior Nodes in Vehicular Ad-Hoc Networks Using Hash Function: A Survey," *International Journal of Engineering and Technical Research (IJETR),*vol. 8, no. 6, pp. 48-52, 2018.

[16] Rens W. van der Heijden, et al., "VeReMi: A Dataset for Comparable Evaluation of Misbehavior Detection in VANETs," *International Conference on Security and Privacy in Communication Systems. Springer,* 2018, pp. 318-337.

[17] Tong Liu, et al., "Naive Bayes Classifier Based Driving Habit Prediction Scheme for VANET Stable Clustering," *Mobile Networks and Applications,* vol. 25, no. 25, pp. 1708-1714, 2019.

[18] M. Mendes Faria and A. Maria Monteiro, "Intrusion Detection in Computer Networks based on KNN, K-Means++ and J48," *Proceedings of SAI Intelligent Systems Conference. Springer,* 2019, pp. 256-271.

[19] Saud Mohammed Othman, et al., "Intrusion detection model using machine learning algorithm on Big data environment," *Journal of Big Data,* vol. 5, no. 1, 2018.

[20] Leon Bottou, "Large-Scale Machine Learning with Stochastic Gradient Descent," *Proceedings of COMPSTAT'2010. Physica-Verlag HD,* 2010, pp. 177-186.

[21] Thomas M. Hehn, et al., "End-to-End Learning of Decision Trees and Forests," *International Journal of Computer Vision,* vol. 128, pp. 997-1011, 2020.

[22] Gyanendra Chaubey, et al., "Thyroid Disease Prediction Using Machine Learning Approaches," *Natl. Acad. Sci. Lett.* 2019.

[23] Prashil Negandhi, et al., "Intrusion Detection System Using Random Forest on the NSL-KDD Dataset," *Emerging Research in Computing, Information, Communication and Applications, Springer,* pp. 519-531, 2019.

[24] Raghavendra S. and Santosh Kumar J., "Performance evaluation of random forest with feature selection methods in prediction of diabetes," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 1, pp. 353-359, 2020.
[25] Girish S., et al., "Mining the Web Data for Classifying and Predicting Users' Requests," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 4, pp. 2390-2398, 2018.
[26] David H. Wolpert and William G. Macready, "No Free Lunch Theorems for Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67-82, 1997.
[27] Faud A. Gahleb, et al., "An Effective Misbehavior Detection Model using Artificial Neural Network for Vehicular Ad hoc Network Applications," *IEEE Conference on Application, Information and Network Security (AINS)*, 2017, pp. 13-18.
[28] B. M. Bidgoli, et al., "Performance Evaluation of Decision Tree for Intrusion Detection Using Reduced Feature Spaces," *Trends in Intelligent Systems and Computer Engineering*, vol. 6, pp. 273-284, 2008

BIOGRAPHIES OF AUTHORS

Abhilash Sonker received his Bachelor degree (B.E.) from SATI, Vidisha in 2006,Master degree (M.Tech) from MANIT,Bhopal in 2009. He is currently Assistant Professor in Information Technology Department at Madhav Institute of Technology & Science, Gwalior, India. His current research interests include Mobile Adhoc Ne twork and Network Security.

R. K. Gupta received his Ph.D. Degree from IIITM Gwalior. He is currently Professor in Computer Science Engineering Department at Madhav Institute of Technology & Science, Gwalior, India. His current research interests include Data Mining, Image Processing & Mobile Adhoc Network.