Clustering Algorithm and Ensemble SVM for Attack Detection in VANET

A. Mythili and S.K. Magendran

1Bharathiar University, Coimbatore, India.
mythili.a@grd.edu.in

2Ooty Government Arts College, Nilgiris, India.
sk.mahendran@yahoo.co.in

Abstract

A Vehicular Ad Hoc Network (VANET) consists of a group of vehicles that communicate with each other without any pre-existing management infrastructure through wireless transmission. This information exchange is not always reliable due to the constraints of malicious users intended forging information to serve their interests. But how to detect multiple attacks in the VANET and how to provide more security while satisfaction of quality of services in VANET becomes a major difficult task. The proposed model consists of four main steps: Data Collection, Data Exchange, Data Analysis and Data Propagation. At data collection stage, cluster head selection is performed by using Ant Colony Optimization (ACO), maintenance of cluster size, avoiding cheating nodes, avoiding link failures and load balancing. Once the cluster heads are selected then communication between one to another node is performed with secure clustering. Once cluster head are selected then data exchange is performed here attack details are recorded by sending hello messages. Then data analysis and data propagation is performed by using Ensemble SVM Classifier to detect attacks. This classifier acts in real-time by monitoring the network activity using graphic representations to detect any abnormal behavior during the communication process.

Key Words: VANET, attack detection, optimization, clustering, ensemble, SVM classifier.
1. Introduction

Nowadays, due to an exponential increase in vehicles on the road, the number of road hazards has grown. Every week, lakhs of people lose their life [1]. Providing secure and reliable driving environment is important. To provide this environment, Vehicular Ad Hoc Networks (VANETS) is used. This mechanism allows the vehicles can interact with each other. It is one of the type of Mobile Ad Hoc Network, (MANET) where vehicles are considered as nodes each vehicle is equipped with transmission capabilities that are interconnected to form a network. VANET is divided into three categories. In the first category Wireless Wide Area Network (WWAN), access points of the cellular gateways are fixed. Access points of the WWAN are used at specific points in the second category known as Hybrid Wireless Architecture. The third category is the Ad Hoc V2V Communication which does not require any fixed access points.

In spite of the numerous benefits offered by VANETs, VANETs are unprotected from multiple threats and attacks. There can be several malicious drivers who send false information or flood the network making the network unavailable for some time. Some of the attacks include Denial-of-service attack, ID Disclosure Attack, Sybil Attack and Sending Illusory Message Attack. Denial-of-service attack (DoS attack) is one type of cyber-attack in which intruder seeks to establish a network resource unavailable to its intended user by temporarily or permanently by disrupting services of a host connected to the Internet. Denial of service is typically accomplished by flooding the targeted machine or resource with redundant requests. This is an attempt to overload systems and prevents some or all legitimate requests from being fulfilled. Another way of launching this attack is by crashing the communication channel. The author in [2] proposes a method to combat DOS.

In ID Disclosure Attack, the capability of the node to hide its identity is broken, and as a result of its location also becomes visible to the whole network. The intruder injects malware into the neighbors of the target node. This malware is replicating in nature and hence targets its neighbors. When the malware reaches the neighbor of the intruder, it notes the location of target node as well as its identity is indicated [3]. Sybil Attack In this type of attack, duplicate nodes is formed by taking up the identities of the other nodes or by using illegal identities. Therefore, sending information to other nodes using different identities.

As a result, different nodes have a different impression about the same node. There are various techniques to handle this attack. The author in [4] proposes a method to control the Sybil attack in VANETS using statistical and probability approach. Sending Illusory Message Attack, In this attack, a node purposely sends false messages like a traffic jam or road accident to another node in the network to create chaos. The authors of [5] proposed various techniques to detect the satisfied node [5]. The rest of the paper is organized as follows.
Section 2 discusses the literature review of VANET and several algorithms for attack detection VANET. Section 3 overviews the proposed detection technique for multiple attacks. Experimental results of the proposed scheme are presented in Section 4. Concluding remarks with future work are covered in Section 5.

2. Literature Review

Diverse network concerns and security challenges is part of VANETs. It is due to get the availability of ubiquitous connectivity, secure communications, and reputation management systems. This affect the trust in cooperation and negotiation between mobile networking entities. In [6], detailed survey is presented. The security features, challenges, and attacks of VANETs, and classification of the security attacks of VANETs due to the different network layers are surveyed. Decentralized VANET provides communication among vehicles and is. The main aim of VANET is providing road safety, enhancing traffic efficiency but it is a network, so VANET also have challenges about security and is prone to attack. To tackle these attacks, the authors of [7] introduced the concept of Intrusion detection system. Also presented the some security attacks and some of the techniques in conjunction with IDS and comparison between them.

The authors of [8] presented a novel cyber-attack classification approach using improved Support Vector Machine (iSVM) by modifying Gaussian kernel. Class specific Cyber Attack Detection System which combines feature reduction technique and improved support vector machine classifier. This technique has two phases, in the first phase we reduced the redundant features of the original KDDCUP2009 dataset by Generalized Discriminant Analysis (GDA). In the second phase used improved Support Vector Machine (iSVM) classifier to classify the reduced dataset obtained from first phase. Result shows that iSVM gives 100% detection accuracy for Normal and Denial of Service (DOS) classes and comparable to false alarm rate, training, and testing times.

An ensemble of Ada Booster with SVM (EAB-SVM) classifier technique is proposed in [9] for detecting the network intrusions and monitoring the activities of the node as well as classifying it as either normal or anomalous. The optimal feature selection method is applied in EAB-SVM to select the relevant features to classify and detect the intrusion in wireless ad-hoc network. After that, an ensemble of Ada booster with SVM (EAB-SVM) classifier is used for categorizing the intrusion through updating the weight of samples. Finally, the objective function of the EAB-SVM classifier is used to distinguish the anomalous and normal node behavior accurately. This, in turn, improves the anomaly intrusion detection accuracy. The simulation results show that the EAB-SVM technique achieved the better performance regarding packet delivery ratio, classification time, false positive rate and anomaly intrusion detection accuracy compared to state-of-the-art methods. The authors of [10] proposed a Cyber Attack Detection System (CADS) and its generic framework, which performs well for all the classes. This thesis based on Generalized Discriminant
Analysis (GDA) algorithm for feature reduction of the cyber-attack dataset and an ensemble approach of classifiers for classification of cyber-attacks. The ensemble approach of classifiers classifies cyber-attack based on the union of the subsets of features. Thus, it can detect a broader range of attacks. The C4.5 and improved Support Vector Machine (iSVM) classifiers are combined as a hierarchical hybrid classifier (C4.5-iSVM) and an ensemble approach combining the individual base classifiers and hybrid classifier for best classification of cyber-attacks. The experimental results illustrate that the proposed Cyber Attack Detection System is having higher detection accuracy for all classes of attacks with minimizing training, testing times and false positive alarm.

3. Proposed Methodology

At the initial stage of the work, selection of cluster head using ant colony optimization (ACO) is used. Finally, a novel Ensemble SVM Classifier is proposed to detect the attacks utilizing a quality. The new method will provide better detection rate for various attacks. The proposed model consists of four main steps: Data Collection, Data Exchange, Data Analysis and Data Propagation.

**Data Collection Algorithm**

The CMs continuously monitor and analyze MPR nodes and collect data (features) regarding their behavior. In the data collection algorithm, first all the necessary parameters are initialized. Then procedure allows collecting the data.

```
Algorithm 1 Data collection.
1: Initialization:
2: Let CH(c) be the cluster-head of cluster C.
3: Let MPRSet(C) be the set of elected MPRs in cluster C.
4: Let WatchdogSet(C) be a the set of watchdogs in cluster C.
5: Let PacketSet(m) be a the set of packets to be sent by the MPR m.
6: Let s be a packet sent by a source node to an MPR.
7: Let count/PacketsToSend(m) be the number of packets to be sent by the MPR m.
8: Let count/SentPackets(m) be the number of packets actually sent by the MPR m.
9: Let count/DroppedPackets(m) be the number of packets dropped by the MPR m.

10: procedure DATA COLLECTION
11: for each MPR m ∈ MPRSet(C) do
12: for each packet p ∈ PacketSet(m) do
13: increment count/PacketsToSend(m)
14: if p = s then
15: increment count/SentPackets(m)
16: else
17: increment count/DroppedPackets(m)
18: end if
19: end for
20: if count/SentPackets(m) = count/PacketsToSend(m) then
21: initializeClass(m) = malicious
22: else
23: initializeClass(m) = cooperative
24: end if
25: end for
26: end procedure
```

Figure 1: Data Collection Algorithm
Data Exchange

All the data are shared in this data exchange phase. The detailed initialization and procedure given in the following figure.

```
Algorithm 2 Data exchange.
1: Initialization:
2: Let watchdogs(C) denote the set of watchdogs in cluster C.
3: Let w be a watchdog vehicle in watchdogs(C) running this algorithm.
4: Let Observations(x) be the set of observations collected by the watchdog x.
5: procedureDataExchange
6: for each watchdog \( \tilde{w} = \text{watchdogs}(C) \setminus \{w\} \) do
7: Broadcast HELLO message to \( \tilde{w} \)
8: Broadcast the observations set to \( \tilde{w} \), i.e.,
9: \( \text{Observations}(\tilde{w}) := \text{Observations}(\tilde{w}) \cup \text{Observations}(w) \)
10: end for
end procedure
```

Figure 2: Data Exchange Algorithm

Data Analysis

This phase uses the data collected to train the enhanced ensemble SVM and classifies the MPR as cooperative or malicious. This allows predicting the final classes of the test.

```
Algorithm 3 Data analysis.
1: Initialization:
2: Let watchdogs(C) denote the set of watchdogs in cluster C.
3: Let w be a watchdog vehicle in watchdogs(C) running this algorithm.
4: Let watchdogs(C) \( \setminus \{w\} \) denote the set of all watchdogs in watchdogs(C) except for w.
5: Let observations be the observations collected by the watchdog w.
6: Let train be the training set of the watchdog w.
7: Let test be the test set of the watchdog w.
8: Let SupportVectors be the support vectors used by the classifier of watchdog w to distinguish among classes.
9: Let MPRSet(C) denote the set of MPRs elected in cluster C.
10: procedureDataAnalysis
11: for each MPR \( m \in \text{MPRSet}(C) \) do
12: \( \text{train} := \text{SupportVectors} \cup \text{observations}(\text{watchdogs}(C) \setminus \{w\}) \)
13: \( \text{test} := \text{observations} \)
14: Train the classifier to find the hyperplane that maximizes the margin between classes in train by solving Eq. (7).
15: Test the classifier on test by pairing each set of inputs with the expected output.
16: Use the learned classifier C to predict the final classes of test.
17: end for
end procedure
```

Figure 3: Data Analysis Algorithm

Data Propagation

The cluster head propagates the classes determined within the cluster to other clusters. The detailed initialization and data propagation process is given in the following figure.
The proposed methodology applies optimized ensemble creation using Hybrid Adaboost and random subset sampling algorithm to the collected features. Finally the malicious nodes are detected using enhanced ensemble SVM classifier.

Algorithm 4 Data propagation.

1: Initialization:
2: Let $C_1$ and $C_2$ be two different clusters.
3: Let $x$ be a cluster head of cluster $C_1$.
4: Let $y$ be a cluster head of cluster $C_2$.
5: Let $\text{members}(x)$ be the members in the cluster led by the cluster-head $x$.
6: Let $\text{members}(y)$ be the members in the cluster led by the cluster-head $y$.
7: Let $\text{maliciousSet}(x)$ be the set of vehicles classified as malicious within $C_1$.
8: Let $\text{maliciousSet}(y)$ be the set of vehicles classified as malicious within $C_2$.
9: Let $\text{cooperativeSet}(x)$ be the set of vehicles classified as cooperative within $C_1$.
10: Let $\text{cooperativeSet}(y)$ be the set of vehicles classified as cooperative within $C_2$.

11: procedure DATA PROPAGATION
12: if a contact between $x$ and $y$ happens then
13: \quad $\text{maliciousSet}(x) := \text{maliciousSet}(x) \cup \text{maliciousSet}(y)$
14: \quad $\text{cooperativeSet}(x) := \text{cooperativeSet}(x) \cup \text{cooperativeSet}(y)$
15: end if
16: for each vehicle $i \in \text{members}(x)$ then
17: \quad $\text{maliciousSet}(i) := \text{maliciousSet}(i) \cup \text{maliciousSet}(x)$
18: \quad $\text{cooperativeSet}(i) := \text{cooperativeSet}(i) \cup \text{cooperativeSet}(x)$
19: end for
20: for each vehicle $j \in \text{members}(y)$ then
21: \quad $\text{maliciousSet}(j) := \text{maliciousSet}(j) \cup \text{maliciousSet}(y)$
22: \quad $\text{cooperativeSet}(j) := \text{cooperativeSet}(j) \cup \text{cooperativeSet}(y)$
23: end for
24: end procedure

Figure 4: Data Propagation Algorithm

Proposed Enhanced SVM Classifier

The proposed methodology applies optimized ensemble creation using Hybrid Adaboost and random subset sampling algorithm to the collected features. Finally the malicious nodes are detected using enhanced ensemble SVM classifier.
15 classifiers created using the extracted features. For Level 1, classifiers such as SVM, KNN and enhanced SVM are used. To create the ensemble, Hybrid Adaboost and random Subspace selection Technique (HAR) is used. Hold-out partitioning method and majority voting aggregation methods are utilized for the voting process. Training of the individual classifiers and applying aggregation does not require further training.

**Ensemble Creation using HAR Algorithm**

Input: Training examples, K, Dimension of the subspaces p*

Output: Ensemble E

\[
E \leftarrow 0
\]

For \( i = 1 \) to \( B \) do

\( S_i = \text{BootstrapSample}(S) \)

\( S'_i = \text{SelectRandomSubspace}(S_i, p^*) \)

\( C_{Li} \leftarrow \text{ConstructClassifier}(S'_i) \)

\( E = E \cup C_{Li} \)

End for

Return E

**Support Vector Machine (SVM)**

SVM uses the quadratic programming for identifying the supports vectors. When the number of samples in the training set is high, determining of potential SVs is difficult. So by reducing the number of SVs used during classification has a direct impact on the speed of SVM. The proposed algorithm uses this concept and removes irrelevant SVs to speed up classification task. The procedure for proposed SVM is given in this slide. Perform pre-clustering of training dataset using K-Means algorithm and obtain clusters along with cluster centers. Identify crisp clusters, which is combined with a threshold to identify irrelevant SVs.

\[
\text{Threshold samples percentage} = \frac{\text{sample within the inner cluster}}{\text{sample within the outer cluster}} \times 100
\]

**Algorithm to Identify Crisp Clusters**

Step 1: For all the cluster centers, increase the cluster radius and assign label as “undecided”.

Step 2: Select a cluster with the label ‘undecided’, \( t=1 \).

If no samples with label ‘undecided’, stop.

Step 3: If all the samples are in same class, increase the radius by \( R \) and check again (\( n=n+1 \)).

Repeat this step until the radius is big enough to include a combination of different class samples.

Select the cluster with previous radius and check all samples in same class.
Assign it as a ‘crisp’ cluster. Go to Step 2, t = t + 1.

**Two-Level Classification Process**

The two-level classification process is illustrated in the figure. The classification task is divided into two levels. In the first level, algorithms such as KNN, SVM and enhanced SVM are applied to the training dataset. In the level 2 task, algorithms are applied to the test dataset. Finally, classification results are produced.

To reduce the complexity pruning algorithm is used. Pruning produces a small set of classifiers that provides the same benefits as the original unpruned ensemble system. In the existing system, the static method uses criterion to select the best classifier and dynamic method uses pruning of classifiers. Here time complexity is the primary issue.

**Proposed Solution and Contribution**

Criteria-based Dynamic Scheduling Pruning (CDSP) is introduced with the contribution of Hybrid Static and Dynamic Pruning. It consists of 3-step Procedure. All the three steps are designed to accelerate the process of classification further. The objective here is to identify an ensemble of classifiers containing only a small set of k classifiers that produces the same benefits as the original unpruned ensemble system.

Steps 1: Minimum error criteria to select classifiers (Static).
Step 2: Maximum benefit criteria to select classifiers (Static).

Step 3: Uses a scheduling method to prune classifiers (Dynamic).

**CDSP-Pruning Algorithm**

Step 1: Mean Square Error Criterion: It aims to identify those classifiers that minimize classification error using Mean Square Error Criterion

Step 2: Total Benefit: Defined as the amount of correctly detected nodes minus the cost to investigate all nodes predicted as malicious (Both correctly and incorrectly). A set of base classifiers are obtained that reduce error and increase the total benefit.

Step 3: Arrange the resultant set of classifiers based on their overall benefits into a “pipeline”. The classifier with the highest benefits will always be consulted first, followed by classifiers with decreasing total benefits. Then stops as soon as “a confident prediction” is made or there are no more classifiers in the pipeline. The confidence is calculated as the ratio of errors to the estimated probability of the original ensemble system.

4. **Experimental Results**

The performance comparison is performed with NS2 simulation environment.

**Performance Metrics**

To evaluate the performance of the proposed algorithm, four performance metrics accuracy rate, attack detection rate, false positive rate and packet delivery ratio are used.

Accuracy Rate=100% * (Total Number of Correctly Classified Processes/Total Number of Processes)

Attack Detection Rate=100% * (Total Number of Attacks/Total Number of Detected Attacks)

False Positive Rate=100% * (Total Number of Misclassified Process/Total Number of Normal Process)

Packet Delivery Ratio= Total Number of Received Packets/Total Number of Sent Packets

<table>
<thead>
<tr>
<th>Metric</th>
<th>SVM</th>
<th>Enhanced SVM</th>
<th>Proposed Ensemble SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy (%)</td>
<td>97.46</td>
<td>98.01</td>
<td>98.99</td>
</tr>
<tr>
<td>Attack Detection (%)</td>
<td>95.43</td>
<td>96.89</td>
<td>97.75</td>
</tr>
<tr>
<td>False Positive Rate (%)</td>
<td>1.4031</td>
<td>0.8776</td>
<td>0.8425</td>
</tr>
<tr>
<td>Packet Delivery Ratio (%)</td>
<td>86.12</td>
<td>91.26</td>
<td>95.43</td>
</tr>
</tbody>
</table>
Performance of Standard Support Vector Machine (SVM), Enhanced SVM and proposed novel Ensemble SVM are evaluated against the four performance metrics. Proposed ensemble SVM gives the accuracy of 98.99% which outperforms existing algorithm performances are illustrated in figure 7 and discussed in table 1. New algorithm significantly reduces the false positive rate. Also, attack detection percentage, and packet delivery ratio is high in proposed ensemble SVM than existing SVMs.

Proposed ensemble SVM gives the accuracy of 95.43% which outperforms existing algorithm performances are illustrated in figure 8 and discussed in table 1.
5. **Conclusion**

In this paper, an enhanced SVM based ensemble algorithm which optimized both SVM operation and ensemble system was proposed. The experimental result proved that the proposed algorithm was efficient in identifying malicious nodes. Support Vector Machine algorithm provides the accuracy of 97.46%, while Enhanced SVM gives 98.01%. But advanced optimization-based clustering algorithm and Ensemble SVM classifier outperforms existing algorithms by providing 98.99% of accuracy for multiple attack detection in VANET. So, it can be concluded that the proposed algorithm is efficient in securing the clustered VANET.

**References**


