Noname manuscript No.
(will be inserted by the editor)

Machine Learning-driven Optimization for SVM-based Intrusion Detection System in Vehicular Ad Hoc Networks

Ayoub Alsarhan · Mohammad Alauthman · Esra'a Alshdaifat · Abdel-Rahman Al-Ghuwairi · Ahmed Al-Dubai ·

Received: date / Accepted: date

Abstract Machine Learning (ML) driven solutions have been widely used to secure wireless communications Vehicular ad hoc networks (VANETs) in recent studies. Unlike existing works, this paper applies support vector machine (SVM) for intrusion detection in VANET. The structure of SVM has many computation advantages, such as special direction at a finite sample and irrelevance between the complexity of algorithm and the sample dimension. Intrusion detection in VANET is nonconvex and combinatorial problem. Thus, three intelligence optimization algorithms are used for optimizing the accuracy value of SVM classifier. These optimization algorithms include Genetic algorithm (GA), Particle Swarm Optimization (PSO), and Ant Colony Optimization (ACO). Our results demonstrate that GA outperformed other optimization algorithms.

Keyword: Intrusion detection, Smart city, support vector machine, security, misbehavior detection.

Ayoub Alsarhan
Department of Computer Information System, Hashemite University, Jordan. E-mail: ayoubm@hu.edu.jo

Mohammad Alauthman
Department of Information Security, Faculty of Information Technology, University of Petra, Amman, Jordan. E-mail: mohammad.alauthman@uop.edu.jo

Esra'a Alshdaifat
Department of Computer Information System, Hashemite University, Jordan. E-mail: esraa@hu.edu.jo

Abdel-Rahman Al-Ghuwairi
Department of Computer Information System, Hashemite University, Jordan. E-mail: ghuwairi@hu.edu.jo

Ahmed Al-Dubai
School of Computing; Edinburgh Napier University, Edinburgh, UK. E-mail: A.Al-Dubai@napier.ac.uk

1 Introduction

Nowadays, smart vehicles have been a corner stone in the concept of intelligent traffic systems (ITS), playing key roles in enhancing the road safety and improving the driving experience. For smart city, road infrastructure relies on information and communications technologies where smart vehicles are connected with other vehicles and roadside units (RSU). ITS relies on RSU for reducing accidents and improving driving performance Alsarhan et al. (2018); Arif et al. (2019).

VANETs are vulnerable to several types of attacks due to the nature of exposure of wireless medium Arif et al. (2019). Unfortunately, these attacks degrade the performance of VANET significantly and create serious problems for legitimate drivers. Hence, protecting VANET's traffic from modification, interception, and deletion the messages has been a great challenge and among the top priorities for academia and industry. Malicious nodes may alert the messages which are used to guide the drivers. Furthermore, attackers may spread false messages which may cause accidents Arif et al. (2019); Sun et al. (2019). In order to apply the VANETs in ITS effectively, new algorithms for securing VANETs' traffic should be proposed and applied efficiently. All VANETs' traffic should be classified for trustworthiness. Every attack class in VANET has certain characteristics, which establishes a profile for it. Machine learning (ML) algorithms have been widely adopted for analyzing large amounts of data and extract useful rule for the detection, classification, and prediction of future events in VANET. ML has many applications specially for problems that involve pattern recognition such as intrusion detection in VANET Alshammari et al. (2018). ML-based security system is desirable solution in such large wireless network so that intrusion can be detected in advance with acceptable time and reasonable accuracy Alshammari et al. (2018). Many factors motivate using ML for intrusion detection, including the following ones.

- The growing availability of large amounts of data, where, RSUs and vehicles can be used to collect data.
- The dramatic advances in ML algorithms such as SVM.
- The emergence of new computing paradigms such as cloud computing, fog computing and edge computing, graphics processing unit (GPU) technology, and other hardware upgrades.

ML can be used to identify unusual patterns of behavior in VANET data which become increasingly voluminous. Several ML algorithms have been used for intrusion detection system (IDS) in VANET Alshammari et al. (2018). These algorithms include: Bayesian approach, Neural network, Decision trees, and SVM.

Bayesian approach is a statistical technique based on Bayesian theorem Fouladi et al. (2016). Naive Bayes classification assumes all features for each event in VANET are independent from each other. It is easy to implement when the dimensionality of the objects is high with high accuracy value for classification. It can be used easily for very large datasets. Usually, it outperforms most of classification methods despite its simplicity Fouladi et al. (2016). In Naive Bayes classification, Bayes theorem is used to compute the posterior probability using large dataset Fouladi et al. (2016). The classifier can handle missing data efficiently Fouladi et al. (2016). For this classifier, it is assumed that the effect of the value for any feature on class C is independent of the values of the other attributesFouladi et al.

(2016). However, the conditional independence of data features is not valid assumption for intrusion detection problem in VANET. Correlated features for each event in VANET may degrade the classification accuracy of IDS significantly.

Artificial Neural Networks (ANN) can handle both linear and non-linear patterns. The extracted model computes the probability with features matches the characteristics that it has been trained to recognize Saied et al. (2016). ANN can learn efficiently from its environment. However, ANN does not scale easily when the data size is large and the structure of ANN is complex.

Decision tree (DT) is non-parametric data mining method used for supervised classification. DT performs a successive partitioning of cases until all objects in dataset are classified. DT can generate a rich set of rules that are easy to integrate with any real time system Gupta et al. (2019). It has been widely used in IDS Gupta et al. (2019). Beside it requires more time for training the model, small variations in the data may change DT structure significantly Gupta et al. (2019).

SVM has become one of the popular ML algorithm that used in IDS due to high generalization performance, and its ability to handle a small sample size Vapnik (1999). Furthermore, it can efficiently deal with curse of dimensionality Vapnik (1999). SVM maps training data nonlinearly into higher-dimensional feature space using mapping function Vapnik (1999). In this work, we use three intelligent algorithms to optimize accuracy function for IDS. These algorithms include: GA Goldberg (2006); Sivanandam and Deepa (2008), ACO Zhang et al. (2018), and PSO Eberhart and Kennedy (1995).The proposed IDS integrates machine learning intelligence and ambient intelligence concepts in VANET to develop decentralized security system that responds to the requirements of road safety. This system enables Vehicles which are considered to be more intelligent to detect any intrusion in VANET by setup SVM-based IDS.The main contributions of this work can be summarized as follows:

1. Propose new SVM-based IDS enhanced with a new penalty function for reinforcing the regularization of the proposed classifier. Penalty function is used to control the complexity of the classifier by reducing the number of support vectors.
2. Since SVM-based IDS depends on several parameters, three algorithms were studied for automatically tuning the parameters of IDS. These algorithms include: PSO, GA, and ACO. New objective function is defined for SVM and the optimizations algorithms are used for the optimization task with certain heuristics in order to avoid trapping into local optimum.
3. Implement the proposed SVM model and examine the model by doing experiment on a well-known intrusion detection dataset (i.e. NSL-KDD).

The remainder of this paper is organized as follows. Section 2 gives a brief overview of the using machine learning based methods for intrusion detection. Section 3 introduces a description of the SVM optimization using intelligent algorithms. The performance result of our scheme is evaluated in Section 4. Section 5 concludes the paper and discusses future work.

2 BACKROUND

ML has a wide range of mundane and complex applications including IDS. Several ML algorithms for building IDS have been proposed. SVM was proposed in

Theissler (2014) for IDS with the Radial Basis Function (RBF) to classify the behaviors of nodes. However, the classifier gives good results in some environments but it fails to detect most anomalies as expected. Neural network was used in Ludwig (2017) for classifying events in the network. The proposed IDS consists of auto encoder, deep belief neural network, deep neural network, and an extreme learning machine. The NSL-KDD data set was used to evaluate the classification performance. Convolutional neural networks (CNN) was proposed in Wu et al. (2018) for intrusion detection. CNN was used to select traffic features from raw data set automatically. Furthermore, cost function was used to set weight coefficient of each class based on its numbers to solve the imbalanced data set problem. Deep learning theory was used in Xu et al. (2018) for improving the performance if IDS and extracting features for each event. The proposed technique has shown good performance. SVM was used in Mohammed and Sulaiman (2012) to improve intrusion system by recognizing attack patterns from database. The data was originated from a computer Lab. New ML based algorithm was applied for intrusion detection in Feng et al. (2014). The proposed algorithm was used to classify network activity as normal or abnormal using the network log. Ant Colony Network was applied with SVM for activities classification. The algorithmes were implemented and evaluated using a standard benchmark KDD99 data set.

Authors in Shone et al. (2018) proposed a new novel deep learning technique for intrusion detection. Nonsymmetric deep learning was used for classifying events. The proposed classifier has been implemented in graphics processing unit (GPU)-enabled TensorFlow. IDS was evaluated using the benchmark KDD Cup ’99 and NSL-KDD datasets. Novel framework was proposed in Papamartzivanos et al. (2019) for IDS. The framework combines the benefits of self-taught learning and MAPE-K frameworks to produce a scalable, self-adaptive, and autonomous misuse IDS. Deep learning was utilized to improve detection rate by grasping an attack’s nature based on the generalized feature reconstructions stemming directly from the unknown environment and unlabeled data.

In Salama et al. (2011), authors suggested new hybrid scheme that adopted deep belief network and SVM to classify the traffic into two outcomes: normal or attack. The attacks fall into four classes: R2L, DoS, U2R, and Probing. In the proposed scheme, deep belief network was utilized to reduce the dimensionality of the feature sets. Then SVM is used to classify the intrusion into five outcomes: Normal, R2L, DoS, U2R, and Probing. NSL-KDD dataset was used to evaluate the performance of proposed scheme. Authors proposed a two-stage classifier based on RepTree algorithm and protocols subset for IDS Belouch et al. (2017). The scheme was evaluated using the UNSW-NB15 data set and the NSL-KDD data set. In the proposed IDS, the incoming network traffics is divided into three types of protocols TCP, UDP or other. Then, IDS classifies each type into normal or anomaly. After that, a multiclass algorithm is used to classify the anomaly detected in the first phase to identify the attacks class in order to choose the appropriate intervention.

New IDS was proposed in Pozi et al. (2016) to solve the problem of missing rare attacks . Anomalous attacks are detected by SVM-based IDS. GA is used to improve the accuracy of attacks detection. Authors proposed new IDS in Desale and Ade (2015) to reduce the time complexity by selecting only features that are most relevant to the classification modeling problem. The proposed IDS uses mathematical intersection principle for features selection. Furthermore, GA is used for classifying the network traffic. Authors proposed GA for intrusion detection in

Hosseini and Zade (2020). A wrapper technique is used in the proposed IDS for feature selection in the first phase. In the second phase, attacks are detected using an artificial neural network.

In Patel et al. (2015), authors proposed new hybrid IDS where association rule mining and random particle swarm optimization have been applied for attacks detection. The IDS was applied on NSL-KDD dataset. After detecting an intrusion, IDS applies a recheck frame to define the suspicious nodes. After that, association rule is used to extract the associated values which are passed to the next procedure. Authors proposed new learning machine scheme for intrusion detection in Ali et al. (2018) .The parameters for IDS are initialized randomly. PSO is used for optimizing the model. A novel hybrid IDS was proposed in Li et al. (2018) with the purpose of detecting intrusion effectively. Gini index is applied for features selection. Furthermore, the gradient boosted decision tree (GBDT) algorithm is adopted for classifying the traffic of network, and the PSO is used to optimize the parameters of GBDT. Authors proposed new scheme to ensure network security in G et al. (2018).The main concern of the proposed is setting some thresholds on generic based feature selection mechanism for detecting network intrusion. ACO was used to optimize decision tree classifier. Furthermore, ACO was adopted to reduce the data set size and feature selection. ACO was adopted in Alanezi et al. (2013) for data classification. The proposed IDS attempts to classify the network traffic to detect intrusion. The Ant Tree Miner Amyntas classifier was proposed for intrusion detection in Botes et al. (2017) . The proposed IDS uses ACO for feature selection from a data set before inducing Decision Trees that classify network traffic. However, it is worth indicating that some of the presented methods neglected the following:

1. developing an automated, and reliable method to select the values of the parameters for SVM-based IDS to enable high accuracy detection of intrusions. In the literature, the major drawbacks of SVM is optimizing the parameters of the model. The SVM algorithm usually depends on several parameters. However, the tuning objectives of the model are often conflicting. These conflicting objectives include: margin maximization and error minimization.
2. handling the dynamic and complex nature of cyber-attacks in VANET. Unfortunately, more complex attacks are developed and lunched for various purposes.
3. controlling the complexity of SVM model using suitable parameters for regularizing the classifier. Reinforcing the regularization of the SVM classifier is challenging problem especially when it is applied for Margin-maximization. Unfortunately, the training the model for SVM-based IDS requires optimization of the regularization and classifier parameters in order to control the risk of overfitting and the complexity of the boundary. Therefore, new regularized term should be added to SVM model in order to control the complexity of the IDS and the generalization performance of the classifier should be reinforced.
4. developing new mechanism for escaping from local optima solution for intrusion detection problem. The optimization problem for intrusion detection in VANET is not convex. This makes it difficult to find the optimal solution, whereas an optimization algorithm like SVM light will sometimes converges to a local optimum.
5. validating the performance of IDS using NSL-KDD. Some of the presented IDSs used the very old KDD'99 dataset to validate the detection method.

In our work, we have tested the proposed IDS using both datasets (i.e. KDD'99, and NSL-KDD). Furthermore, while we analysis the optimization performance of three ML based algorithms in terms of classification accuracy, most studies neglect the optimization and performance analysis of SVM. In this work, ML-based algorithms are used to look for optimum values for the SVM parameters in all spaces of potential parameters values. For IDS, SVM is simplified by adding new variable to the classification function for controlling the number of generated SVM.

3 OPTIMIZING SVM USING ML ALGORITHMS FOR IDS

SVM separates the training set of vectors that represent the incoming traffic as either malicious or normal. IDS assigns each event in VANET to malicious or normal event based on external observation. Suppose that $x_i \in R^N, 1, ..m$, are m training data that can be classified into two classes, and $y_i \in malicious, normal, i = 1, .., m$, are their corresponding class label. IDS uses a rule for event classification. The problem of intrusion detection can be expressed as follows:

$$f : x_i \rightarrow y_i, i = 1, .., m \tag{1}$$

Let x be the set of test and training examples. The test and training examples are generated from the same probability distribution $P(x, y)$. Assume the training sample is separable by the following hyperplane function (see Figure 1):

$$(\omega.x) + b = 0 \tag{2}$$

where ω is normal to the hyperplane. Figure 1 displays a simple linearly separable case H. H is the optimal separating hyperplane. However, there are an infinite number of these separating hyperplanes. In intrusion detection problem, we have more than one hyperplane which separates the malicious (i.e. positive) from normal (i.e. negative) examples. $H_1$and H_2 are called support vector machines which are the closest to the separating line. The margin boundaries can be specified by drawing lines that pass through H_1 and H_2 which are parallel to H Vapnik (2013). The margin is defined as the distance between H_1 and H_2. IDS based on SVM is optimal if the adopted SVM classifies malicious (i.e. positive) from normal (i.e. negative) examples without error (accuracy rate 100%) and the margin is maximal. The main concern of the detection algorithm of the SVM is maximizing the margin and minimizing the classification error. The margin is computed as follows:

$$M = \frac{2}{||\omega||} \tag{3}$$

IDS attempts to minimize the recognition rate by solving the optimization problem:

$$min_{\omega,b} \frac{1}{2} \omega^T \omega + C \sum_{m}^{i=1} \xi i \tag{4}$$

$$s.t : y_i[\omega^T \phi(x) + b] \geq 1 - \xi_i, \forall_i = 1, ..., m \tag{5}$$

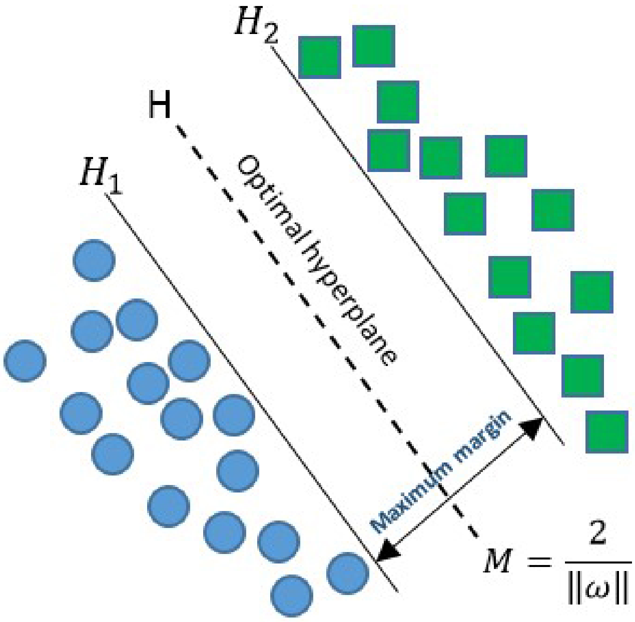

Fig. 1 Optimal Separating Hyperplane

$$\xi_i \geq 0, \forall_i = 1, ..., m \tag{6}$$

where ξi is slack variable, and $\phi(x)$ is a function that map training data into higher-dimensional feature space, which is specified by kernel function $K(x, y)$. The kernel function is evaluated as follows Vapnik (2013):

$$k(x, y) = \phi(x).\phi(y) \tag{7}$$

The Lagrangian for intrusion classification problem is represented as follows Vapnik (2013):

$$\mathcal{L}(W, \propto) = \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{m}\xi_i - \sum_{i=1}^{m}\propto_i [1 - \xi_i - y_i(\omega, \phi(x_i) - b)] - \sum_{i=1}^{m}\pi_i\xi_i \tag{8}$$

We apply the differentiation theorem to obtain Lagrange Vapnik (2013):

$$f(x) = \rho\sum_{i=1}^{s}\propto_i y_i K(xi, x) + b \tag{9}$$

where x_i is one of support vectors, α_i is corresponding Lagrange multiplier, s is the number of support vectors and ρ is a step function. This constraint optimization problem can be expressed as follows Vapnik (2013):

$$min\left|E(\propto^*) - G(\propto, y)\right| - \epsilon \tag{10}$$

$s.t : \sum_{i=1}^{s} \propto_i \; y_i = 0,$
$yi(\omega.\phi(x_i) - b) \geq 1,$
$C \geq \propto_i$

E($\propto^*$) can be expressed as follows:

$$E(\propto^*) = \sum_{i=1}^{s} \propto_i \tag{11}$$

$G(\propto, y)$ can be expressed as follows:

$$G(\propto, y) = \frac{1}{2} \sum_{i=1}^{s} \propto_i \propto_j \; y_i y_j \tag{12}$$

For IDS, the main concern is finding the hyperplane < w,b > that maximize margin and minimize classification error as follows Vapnik (2013):

$$\underset{w, \xi, \xi^*, y_1^*, ..., y_m^*}{min} \mathcal{L}(w, \propto) = \frac{1}{2} \omega^T \omega + C \sum_{i=1}^{m} \xi - C^* \sum_{i=1}^{m} \xi^* \tag{13}$$

$s.t : yi(\omega^T.\phi(x_i) + b) \geq 1 - \xi_i$
$\xi_i \geq 0,$
$y_j^*[(\omega^T.\phi(j) + b)] \geq 1 - \xi_j^*,$
$\xi^* \geq 0,$

According Eq. (13) classification error is minimized by minimizing the classification errors in both training data set and test set Vapnik (2013). Combinatorial optimization approach is used to find the set of labels that minimize classification errors as follows Hoi et al. (2008):

$$\begin{aligned} W(\propto^*) = \sum_{i=1}^{s} \propto_i - \frac{1}{2} \sum_{i=1}^{s} \propto_i \propto_j \; y_i y_j \phi(x_i).\phi(x_i)(+ \\ \sum_{i=1}^{s} \propto_i^* - \frac{1}{2} \sum_{i=1}^{s} \propto_i^* \propto_j^* \; y_i^* y_j^* (\phi(x_i^*).\phi(x_j^*)))- \\ \sum_{i=1}^{s} \sum_{i=1}^{s} \propto_i \propto_j^* \; y_i y_j (\phi(x_i).\phi(x_j^*)) \end{aligned} \tag{14}$$

$s.t : 0 \leq \propto_i \leq C,$
$0 \leq \propto_i^* \leq C^*,$
$\sum_{s}^{i=1} \propto_j^* \; y_j^* = 0,$

It is clear the more number of support vectors leads to more complex decision function. Therefore, our aim is to reduce the number of support vectors by penalizing the increment of the number support vectors. In order to reduce the number of generated support vectors s , we propose the following objective function:

$$g(x) = Zf(x) + (1 - Z)s \tag{15}$$

where ϑ is a real number with $0 \leq Z \leq 1$. ϑ can be expressed as follows:

$$Z = \frac{\vartheta}{\alpha *} \tag{16}$$

with $\vartheta \leq \alpha *$

3.1 PSO for IDS optimization

PSO is a computational method based on the swarm intelligence paradigm, and social behavior of animals. PSO is widely used to solve nonlinear and non-continuous optimization problems Eberhart and Kennedy (1995); Eberhart et al. (2001). Compared with conventional optimization approaches, PSO has less sensitivity to the nature of the objective function. Furthermore, it requires limited number of parameters in contrast with other optimization techniques Eberhart and Kennedy (1995); Eberhart et al. (2001). Particle's position represents a potential solution. The fitness value for each particle is computed using the fitness function. The characteristics for each particle include: position, velocity, and fitness value. Let the feasible solution m -dimensional space, the population of m particles is $X = [x_1, x_2, \ldots, x_m]$. The position and velocity of the i^{th} particle are $P_i = [p_1, p_2, \ldots, p_m]$ and $V_i = [v_1, v_2, \ldots, v_m]$, respectively, where the velocity specifies the direction and distance of i^{th} particle movement. During each iteration, particles move in the feasible solution space for better position. For i^{th} particle, the value of objective function $O(P_i)$ is determined based on the coordinates of the i^{th} particle in the n-dimensional search space which is a certain solution to the optimization problem Sun et al. (2016); Demidova and Sokolova (2015).

After initializing a group of random particles with position and velocity by PSO, velocity and local best location for each particle are updated at iteration K as follows:

$$V_i(k+1) = V_i(k) + c1R()(P_i^*(k)) + c2R()(P^{GB}(k) - P_i(k)) \tag{17}$$

where $V_i(k+1)$ is the velocity vector of the ith particle, c_1 and c_2 are personal and global coefficients for ith particle acceleration. They are constant for determining the behavior and effectiveness of the PSO algorithm. $R()$ is random numbers in interval [0, 1], P_i defining the position of the i^{th} particle; P_i^* is the i^{th} coordinate of the best position vector found by the i^{th} particle. The particle position is updated using the following equation:

$$P_i(K+1) = P_i(k) + V_i(k1) \tag{18}$$

The position for each particle is updated. The PSO algorithm is given as follows:

3.2 IDS Optimization using ACO algorithm

ACO algorithm was proposed in Karaboga (2005) for optimization of an extreme value. Beside it can converge quickly, ACO requires few parameters for extracting optimal solution. ACO has the advantage of inheriting parallelism and solves

Algorithm 1: PSO Algorithm

```
for $(i = 1, ......N)$ do
    $P_i = generateRandomPosition()$;
    $v_i = generateRandomVelocity()$;
end
$fit = bestFit(X)$;
$globalBest = fit$;
$P_{best} = bestPos(X); K = 1$;
while $margine > \epsilon$ do
    for $i$=1,...,$m$ do
        if $(O(P_i) > fit)$ then
            $fit = O(P_i)$;
        end
        $P_{best} = bestPos(X)$; // best position
        if $(O(P_i) > globalBest)$ then
            globalBest= $O(P_i)$;
        end
        $g_{best} = x_i$;
        $V_i(k+1) = V_i(k) + c_1R()(P_i^*(k) - P_i(k)) + c_2R()(P^GB(k) - P_i(k))$
        $P_i(k+1) = P_i(k) + V_i(k+1)$
    end
    $K = K + 1$;
end
```

optimization efficiently Karaboga (2005). ACO algorithm imitates the behavior of a bee colony for gathering high quality nectar. For intrusion detection problem using SVM, the nectar is the optimal separating hyperplane. The fitness function measures the accuracy rate of classifier, where the highest values of fitness function indicates the maximum margin hyperplane. The bee colony consists of three kinds of bees: employed bees, onlooker bees, and scout bees. The employed bees, and onlooker bees become scouts when the quantity of the nectar becomes low. At time t, employed bees generates new solution v_t (i.e. new food source position) that close to the old solutions, $v_(t-1)$ as follows:

$$v_t = v_{t-1} + \varphi(v_t - v_{t-1}) \tag{19}$$

where φ is a random number in the interval [-1,1]. The fitness values of new solutions are calculated as follows:

$$f_t = \begin{cases} \frac{1}{1+O(v_t)}, ifO(v_t) \geq 0 \\ 1 + |O(v_t)|, ifO(v_t) < 0 \end{cases} \tag{20}$$

where $O(v_t)$ is the objective function for the SVM (i.e. maximizing classification accuracy). In order to improve its solution (i.e. maximizing the margin value M), the onlooker bees select employed bees by using (14).

Each unemployed bee selects an employed bee. The probability of selecting i^{th} bee is computed as follows:

$$P(i) = \frac{f(i)}{\sum_N^{j=1} f(i)} \tag{21}$$

where $f(i)$ is the fitness of the i^{th} employed bee. The ACO algorithm is given as follows:

Algorithm 2: ACO Algorithm

```
Generate randomly N separating hyperplanes
while (ClassificationAccuracy > Threshold) do
    for (each employee bee) do
        $v_t = v_{t-1} + \phi(v_t - v_{t-1})$
        $f_t = \begin{cases} \frac{1}{1+O(v_t)}, ifO(v_t) \geq 0 \\ 1 + |O(v_t)|, ifO(v_t) < 0 \end{cases}$
        SelectBestSolution();
    end
    $P(i) = \frac{f(i)}{\sum_N^{j=1} f(i)}$
    for each onlooker bee do
        Select-Best-Solution();
        $v_t = v_{t-1} + \phi(v_t - v_{t-1})$
        f$_t = \begin{cases} \frac{1}{1+O(v_t)}, ifO(v_t) \geq 0 \\ 1 + |O(v_t)|, ifO(v_t) < 0 \end{cases}$
        SelectBestSolution();
    end
    if (there is i employee bee become scout bee) then
        Select-new-source-food(i);
    end
end
```

3.3 Optimizing classification Accuracy of ID using GA

GA Holland et al. (1992) can be used for solving a nonconvex optimization problem such as optimizing SVM. In our work, genetic algorithm is adopted for SVM optimization where the combinatorial approach is used for this problem. We use GA for searching the best labels of hyperplane that maximizes the margins. GA has several advantages over other optimization techniques. These advantages include:

- Parallel search for optimal solution.
- Derivability or convexity is not required for extracting optimal solution. Furthermore, GA adapts to complex problems.
- GA can search the optimal solution in discreet search space. For SVM optimization problem, the labels of each intrusion is discreet.

In intrusion detection problem, GA is used to find the optimal labels that maximize the margin and minimize the classification error. Each event in VANET should be classified into binary string where 0 represents misbehaved event, and 1 well-behaved event. The value of the objective function defined in Eq. (10) is used to evaluate the quality of each candidate solution in the population. For extracting the optimal solution, the selection operator of the GA executes the following tasks:

- Generating the good solution for each population.
- Producing multiple copies of the good solutions.
- Excluding bad solutions.

A set of chromosomes is selected from the current generation's population for producing the next generation's population. Several techniques have been proposed for carrying out the selection of the next generations. In our work, Tournament selection Goldberg (1989) has been adopted for classifying events in VANET. At each iteration, the best solution for the current population is selected. Then, optimal selection is determined using the Tournament technique. In order to generate new solutions in GA, crossover operator is used to combine the gens of one individual with those of another to create new solutions from an existing population. The two- and three-points crossover techniques are used in our work to create new solutions. The mutation operator is used to change a chromosome into its opposite. Hence, a new individual is generated in the population. The criterion that enables mutating only the misclassified samples according to the margin is adopted for intrusion detection problem. Therefore, the probability of choosing a gene is set proportional to the corresponding slack variable ξ. The mutation probability can be expressed as follows:

$$P(m) = P_G \frac{\xi i}{\sum_{j=1}^{N-1} \xi i} \tag{22}$$

where P_G is the probability of selecting the individual solution that owns the gene. Our training scheme attempts to prevent mutation of well classified events. Our scheme starts training the classifier with labeled data for labeling the unlabeled data. If the generated value for x_i of SVM is large (i.e. $f(x_i) > 1$) , the label is kept. The label is chosen randomly for lower value of the predicted label. The chosen heuristics enable GA to generate population that is close to the optimal solution. After that, a new samples of unlabeled data are used to extend the subspace and to optimize the SVM. The GA is repeated until the algorithm converges to a certain value for classification error. The details of GA algorithm are shown in the table below.

Algorithm 3: GA Algorithm

```
Train- SVM-random- samples-data()
while (ClassificationAccuracy > Threshold) do
    Replace-Population ();
    Select-parents();
    Crossover();
    Get-Best-Solution();
end
```

4 SIMULATION AND RESULTS

We simulate the driver and VANET activities, the communication activities in the vehicles and RSUs of the VANET based on the NSL-KDD data. NSL-KDD data is the cleaned-up refined version of KDD'99 Tavallaee et al. (2009). It solves some of the inherent problems of the KDD'99 Tavallaee et al. (2009). NSL-KDD contains all records of KDD dataset. The records contain internet traffic that was handled by a real intrusion detection system. Each record has 43 features where 41 of these features referring to the internet traffic and the last two are labels for the event. Each connection was labeled as either normal or attack. SVM classifier is used in the experiments to classify VANET activities into an attack (1) or normal (-1).

Four types of attacks are presented in NS-LKDD. These attacks include: denial of service, probing, unauthorized access to local system administrator privileges, and unauthorized access from a remote machine Dhanabal and Shantharajah (2015). The NSL-KDD dataset passes through pre-processing phase. In pre-processing phase all necessary steps for speeding up the learning were performed. These steps include: transformation of symbolic attributes to numeric values, feature selection, and data normalization. The following criteria are used for evaluating the performance of IDS:

- True Positive (TP): are intrusion events that correctly classified as abnormal.
- False Positive (FP): are the normal traffic that was incorrectly classified as intrusion.
- True Negative (TN): are normal traffic that correctly classified.
- False Negative (FN): are the anomalous events that were incorrectly classified as normal.

The following metrics are used to evaluate the proposed IDS:

- Accuracy rate: it is the number of correct predictions divided by all number predictions made. It is computed as follows:

$$A_r = \frac{TP + TN}{FP + FN + TP + TN} \tag{23}$$

- Detection rate: the percentage of events that are correctly classified as attacks. It is computed as follows:

$$D_r = \frac{TP}{FP + TP} \tag{24}$$

- False Positive Rate: the percentage of normal events that are incorrectly classified as attacks. It is computed as follows:

$$FP_r = \frac{FP}{TN + FP} \tag{25}$$

- False Negative Rate: the percentage of positive events that are incorrectly classified as negative. It is computed as follows:

$$FN_r = \frac{FN}{FN + TP} \tag{26}$$

The performance of IDS is evaluated based on its capability of classifying VANET traffic into a correct type. 10-fold cross-validation method is used to avoid the effect of data sampling when evaluating the IDS' performance. The experiments were conducted using repeated 10-fold cross-validation method. In our work, all the results are the average value of outputs from 10 iterations of 10-fold cross-validation approach. Firstly, the dataset set is shuffled randomly and divided into 10 groups. One of the datasets is used for testing and the remaining nine groups are used for training. Then, the model is trained on the nine training datasets. After that, the model is validated using the testing dataset. The process repeated

10 times and all the final results reported are the average value of results from 10 iterations of 10-fold cross-validation procedure.Fig. 2 presents a comparison of the achieved rates for the SVM that was optimized with three ML algorithms: GA (GA-SVM), PSO (PSO-SVM), and ACO (ACO-SVM). It is apparent that from Fig. 2 that using GA improves the performance of our IDS by about 4–9%. Clearly, GA-SVM achieves an accuracy rate of 98% which outperforms the other algorithms. In Fig. 3 we have compared the achieved detection rates for the three optimization algorithms. It is seen from the figure that the GA-SVM outperforms the other IDs in term of detection rate. It is apparent that from the figure that use of GA brings up the D_r to values close to 99%.

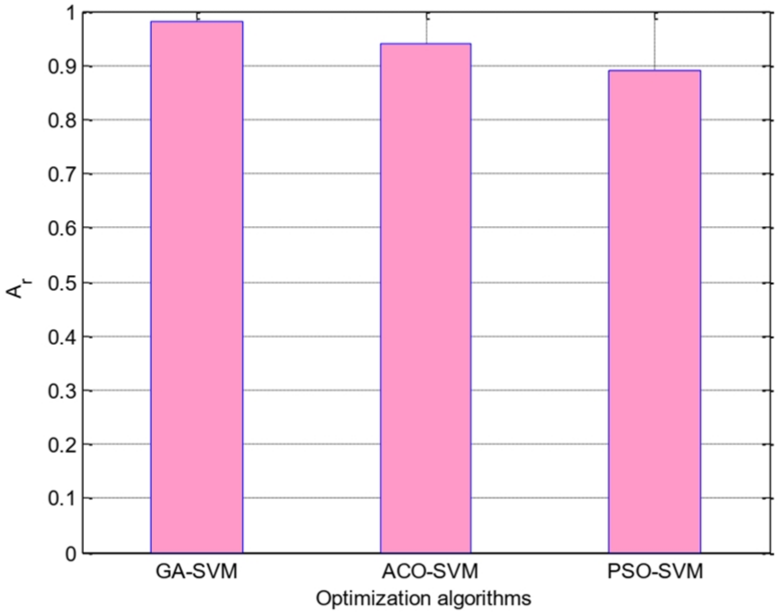

Fig. 2 Accuracy rates comparison for the three optimization algorithms

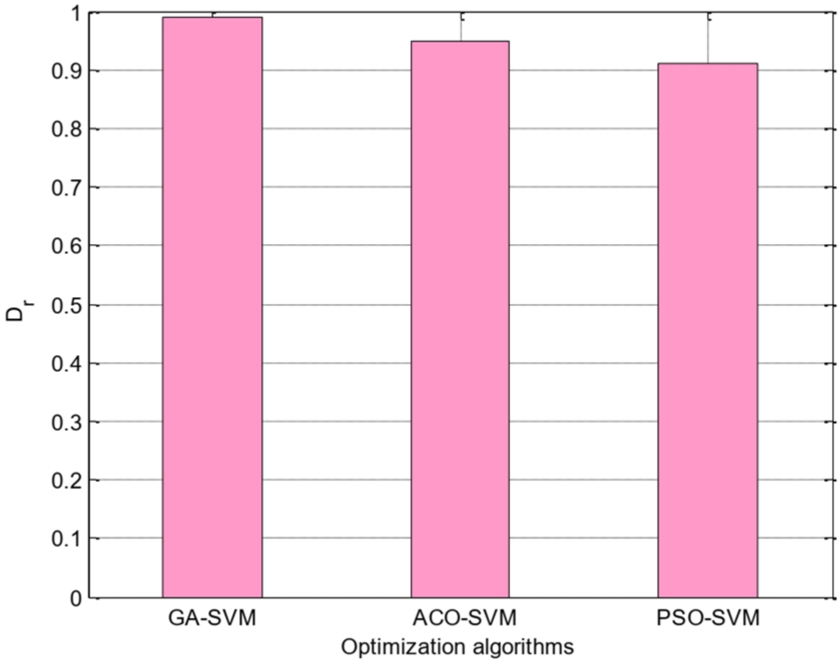

Fig. 3 Detection rates for the optimization algorithms

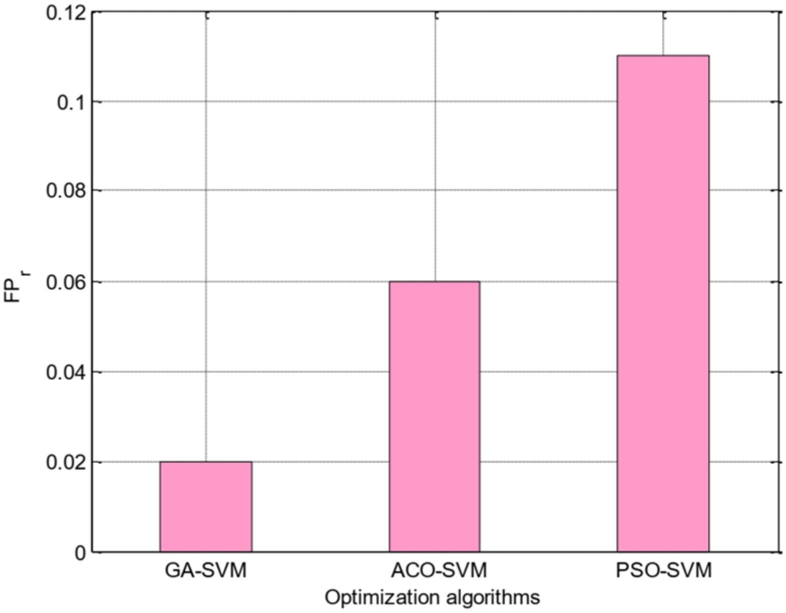

Fig. 4 False positive rates for the optimization algorithms

FP_r indicates the failure in detecting normal events. In other words, false alarm has been raised. FP_r is used to describe the failure of detecting normal behaviors of nodes. Fig. 4 presents a comparison of the reported FP_r for the three optimization algorithms. The GA-SVM technique achieves the least FP% when compared to other IDS.

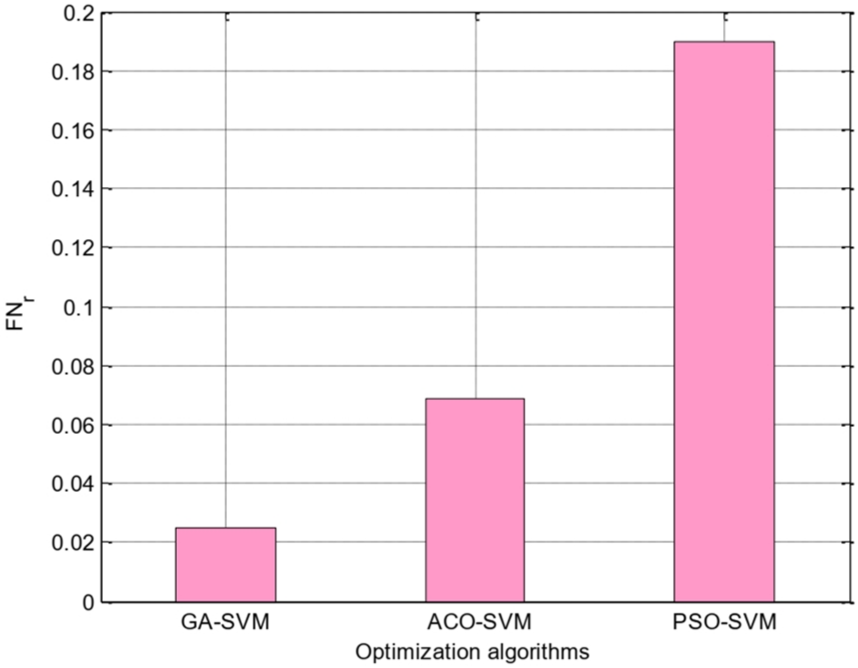

Fig. 5 False negative rates for the three optimization algorithms

False Negative rate (FN_r) refers to the ratio of positive events such as intrusion which is incorrectly classified as negative. Fig. 5 presents a comparison of the FN_r for the three algorithms. The GA-SVM achieves the least FN_r when compared to others algorithms. The experiments illustrate the performance of the proposed IDS. Clearly, the experiments have shown that the proposed classification method has improved the accuracy of the IDS due to the following :

1. adding a penalty factor to the IDS's objective function . Penalty function controls the complexity of the IDS. It prevents the SVM from handling infeasible region and improves the generalization ability of the classifier.
2. Intrusion detection using SVM classifier is combinatorial optimization problem where classifier attempts to produce the optimal hyperplane which separates the malicious (i.e. positive) from normal (i.e. negative) examples. Therefore, GA outperformed other optimization algorithms because this problem is combinatorial optimization problem. Some heuristics are used to avoid local optimum and infeasible solution. These heuristics include: preventing mutation of well classified events, and mutating solely the misclassified samples according to the margin.

5 CONCLUSION AND FUTURE WORK

Smart vehicles provide a promising technology for ITS in smart cities. With the new capabilities of these smart vehicles equipped with other technological advances such as IoT and cloud computing-based ITS applications play a pivotal role in enabling smart cities around the world. However, smart vehicles communication is more vulnerable to various types of cyber-security attacks due to open wireless medium. In order to enable ITS and to prevent such attacks, we analyzed optimizing SVM using three ML algorithms (GA, ACO, and the PSO) for classifying intrusion in NSL-KDD set. We compared the optimization performance of the three ML algorithms on the SVM in terms of classification accuracy. When compared with the ACO and PSO algorithms, the GA outperform its counterparts in terms of classification accuracy. For the near future, we plan to extend the proposed model to utilize new AI-driven models such as deep learning for optimizing SVM and incorporating real big data collected from vehicular communication for SVM training.

References

Alanezi M, Omar B, Abdullah Z, Atallah A (2013) Intrusion detection and classification using ant colony optimization algorithm. In: The 6th Scientific Conference of the College of Computer Sciences Mathematics, Iraqi Journal of Statistical Sciences (25), pp 194–209

Ali MH, Fadlizolkipi M, Firdaus A, Khidzir NZ (2018) A hybrid particle swarm optimization-extreme learning machine approach for intrusion detection system. In: 2018 IEEE Student Conference on Research and Development (SCOReD), IEEE, pp 1–4

Alsarhan A, Al-Dubai AY, Min G, Zomaya AY, Bsoul M (2018) A new spectrum management scheme for road safety in smart cities. IEEE Transactions on Intelligent Transportation Systems 19(11):3496–3506, DOI 10.1109/TITS.2017.2784548

Alshammari A, Zohdy MA, Debnath D, Corser G (2018) Classification approach for intrusion detection in vehicle systems. Wireless Engineering and Technology 9(4):79–94

Arif M, Wang G, Bhuiyan MZA, Wang T, Chen J (2019) A survey on security attacks in vanets: Communication, applications and challenges. Vehicular Communications p 100179

Belouch M, El Hadaj S, Idhammad M (2017) A two-stage classifier approach using reptree algorithm for network intrusion detection. International Journal of Advanced Computer Science and Applications 8(6):389–394

Botes F, Leenen L, de La Harpe R (2017) Ant tree miner amyntas: Automatic, cost-based feature selection for intrusion detection. Journal of Information Warfare 16(4):73–92

Demidova L, Sokolova Y (2015) Modification of particle swarm algorithm for the problem of the svm classifier development. In: 2015 International Conference" Stability and Control Processes" in Memory of VI Zubov (SCP), IEEE, pp 623–627

Desale KS, Ade R (2015) Genetic algorithm based feature selection approach for effective intrusion detection system. In: 2015 International Conference on Computer Communication and Informatics (ICCCI), IEEE, pp 1–6

Dhanabal L, Shantharajah S (2015) A study on nsl-kdd dataset for intrusion detection system based on classification algorithms. International Journal of Advanced Research in Computer and Communication Engineering 4(6):446–452

Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. In: MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science, pp 39–43, DOI 10.1109/MHS.1995.494215

Eberhart RC, Shi Y, Kennedy J (2001) Swarm intelligence. Elsevier

Feng W, Zhang Q, Hu G, Huang JX (2014) Mining network data for intrusion detection through combining svms with ant colony networks. Future Generation Computer Systems 37:127–140

Fouladi RF, Kayatas CE, Anarim E (2016) Frequency based ddos attack detection approach using naive bayes classification. In: 2016 39th International Conference on Telecommunications and Signal Processing (TSP), IEEE, pp 104–107

G P, M J, M S (2018) An optimized decision tree approach for intrusion detection. Eurasian Journal of Analytical Chemistry 13(6):684–688

Goldberg D (1989) Genetic algorithms in search, optimization, and machine learning, addison-wesley, reading, ma, 1989. NN Schraudolph and J 3(1)

Goldberg DE (2006) Genetic algorithms. Pearson Education India

Gupta N, Prasad R, Saurabh P, Verma B (2019) Nb tree based intrusion detection technique using rough set theory model. In: Data, Engineering and Applications, Springer, pp 93–101

Hoi SCH, Rong Jin, Jianke Zhu, Lyu MR (2008) Semi-supervised svm batch mode active learning for image retrieval. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp 1–7, DOI 10.1109/CVPR.2008.4587350

Holland JH, et al. (1992) Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press

Hosseini S, Zade BMH (2020) New hybrid method for attack detection using combination of evolutionary algorithms, svm, and ann. Computer Networks p 107168

Karaboga D (2005) An idea based on honey bee swarm for numerical optimization. Tech. rep., Technical report-tr06, Erciyes university, engineering faculty, computer ...

Li L, Yu Y, Bai S, Cheng J, Chen X (2018) Towards effective network intrusion detection: a hybrid model integrating gini index and gbdt with pso. Journal of Sensors 2018

Ludwig SA (2017) Intrusion detection of multiple attack classes using a deep neural net ensemble. In: 2017 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, pp 1–7

Mohammed MN, Sulaiman N (2012) Intrusion detection system based on svm for wlan. Procedia Technology 1:313–317

Papamartzivanos D, Mármol FG, Kambourakis G (2019) Introducing deep learning self-adaptive misuse network intrusion detection systems. IEEE Access 7:13546–13560

Patel R, Bakhshi D, Arjariya T (2015) Random particle swarm optimization (rpso) based intrusion detection system. International Journal of Advanced Technology and Engineering Exploration 2(5):60

Pozi MSM, Sulaiman MN, Mustapha N, Perumal T (2016) Improving anomalous rare attack detection rate for intrusion detection system using support vector machine and genetic programming. Neural Processing Letters 44(2):279–290

Saied A, Overill RE, Radzik T (2016) Detection of known and unknown ddos attacks using artificial neural networks. Neurocomputing 172:385–393

Salama MA, Eid HF, Ramadan RA, Darwish A, Hassanien AE (2011) Hybrid intelligent intrusion detection scheme. In: Soft computing in industrial applications, Springer, pp 293–303

Shone N, Ngoc TN, Phai VD, Shi Q (2018) A deep learning approach to network intrusion detection. IEEE Transactions on Emerging Topics in Computational Intelligence 2(1):41–50

Sivanandam S, Deepa S (2008) Genetic algorithm optimization problems. In: Introduction to genetic algorithms, Springer, pp 165–209

Sun G, Sun S, Sun J, Yu H, Du X, Guizani M (2019) Security and privacy preservation in fog-based crowd sensing on the internet of vehicles. Journal of Network and Computer Applications 134:89 – 99, DOI https://doi.org/10.1016/j.jnca.2019.02.018, URL http://www.sciencedirect.com/science/article/pii/S1084804519300694

Sun J, Lai CH, Wu XJ (2016) Particle swarm optimisation: classical and quantum perspectives. Crc Press

Tavallaee M, Bagheri E, Lu W, Ghorbani AA (2009) A detailed analysis of the kdd cup 99 data set. In: 2009 IEEE symposium on computational intelligence for security and defense applications, IEEE, pp 1–6

Theissler A (2014) Anomaly detection in recordings from in-vehicle networks. Big data and applications 23

Vapnik V (2013) The nature of statistical learning theory. Springer science & business media

Vapnik VN (1999) An overview of statistical learning theory. IEEE transactions on neural networks 10(5):988–999

Wu K, Chen Z, Li W (2018) A novel intrusion detection model for a massive network using convolutional neural networks. IEEE Access 6:50850–50859

Xu C, Shen J, Du X, Zhang F (2018) An intrusion detection system using a deep neural network with gated recurrent units. IEEE Access 6:48697–48707

Zhang H, Bochem A, Sun X, Hogrefe D (2018) A security aware fuzzy enhanced reliable ant colony optimization routing in vehicular ad hoc networks. In: 2018 IEEE Intelligent Vehicles Symposium (IV), pp 1071–1078, DOI 10.1109/IVS.2018.8500485