Rabindra Bharati University Journal of Economics
ISSN : 0975-802X

MODIFIED FUZZY C MEANS CLUSTERING AND IMPROVED SUPPORT VECTOR MACHINE FOR INTRUSION DETECTION IN VANET

S. DE. Kalaivani, Research Scholar, Park's College, Chinnakkarai.
Dr.A. Nithya, Professor and Head, Department of Computer Applications,Administrative Management College, Bangalore, Karnataka, India.

Abstract
Vehicular Ad-hoc Network (VANET) is a heterogeneous network of resource-constrained nodes such as smart vehicles and Road Side Units (RSUs) communicating in a high mobility environment. Concerning the potentially malicious misbehaves in VANETs, real-time and robust intrusion detection method is required. In existing work introduced random forest and a posterior detection based on coresets to improve the detection accuracy and increase detection efficiency. Increased accuracy requires more trees. However more trees slow down the random forest model. K-means has trouble clustering data where clusters are of varying sizes and density and it is lead to poor results. To avoid these issues in this work proposed an improved model for intrusion detection in VANET. Initially preprocessing is performed to normalize the input data using min max normalization method. Feature selection is done by using Modified Chicken Swarm Optimization (MCSO).Once the feature selection is done it send for signature based intrusion detection using Improved Support Vector Machine (ISVM). Data which are classified as normal traffic in the signature based intrusion detection phase will be send for anomaly based intrusion detection phase. Modified fuzzy c means clustering is used for anomaly based intrusion detection. Proposed model is evaluated interms of accuracy, precision, recall and f-measure.

Keywords: Vehicular Ad-hoc Network, Road Side Units, Min-Max normalization, Modified Chicken Swarm Optimization and Improved Support Vector Machine.

1. Introduction

The Vehicular Ad-hoc Network (VANET) is an emerging type of Mobile Ad hoc Networks (MANETs) with excellent applications in the intelligent traffic system[1]. Despite the promising future of VANETs, they are known to be sensitive to various misbehaves, ranging from malicious attacks to random failures[2]. Considering the safety of vehicles is directly related to human lives, security is one of the main challenges in VANETs. Various detection methods have been proposed in the past decade to detect and mitigate Intrusions in VANETs[3].

Most of these presented methods overlook the security of senior units or just simply rely on a set of predefined and fixed threshold(s) to secure the senior units[4,5]. However, senior units, Road Side Units (RSUs) and Cluster Heads (CHs), are not guaranteed to be safe in a VANET. Although RSUs are built to be robust, yet intruders can still impair the system through physical attacking RSUs or impersonating as an RSU. Not to mention that CHs are easier than RSUs to be impersonated or overtook. The overlook of those senior units' security can lead to serious consequences. Furthermore, considering the highly dynamic nature of VANETs, it is not achievable to find a set of fixed thresholds to detect malicious nodes[6,7].

In contrast, Machine Learning based (ML-based) intrusion detection method can automatically determine whether a node is malicious or not considering all available data from the VANET[8,9]. In existing work introduced random forest and a posterior detection based on coresets to improve the detection accuracy and increase detection efficiency. Increased accuracy requires more trees. However more trees slow down the random forest model. K-means has trouble clustering data where clusters are of varying sizes and density and it is lead to poor results.

To avoid these issues in this work proposed an improved model for intrusion detection in VANET. Initially preprocessing is performed to normalize the input data using min max normalization method. Feature selection is done by using Modified Chicken Swarm Optimization (MCSO).Once the feature selection is done it send for signature based intrusion detection using Improved Support Vector

Machine (ISVM). Data which are classified as normal traffic in the signature based intrusion detection phase will be send for anomaly based intrusion detection phase. Modified fuzzy c means clustering is used for anomaly based intrusion detection.
The rest of this article is divided into 3 sections. The proposed Vanet intrusion detection method is elaborated in Section. 3. The experimental results are shown in Section. 4. Finally, Section. 5 gives the concluding remark of this work and future enhancement.

2. Litrature review

Zeng et al [2018] [10]presented a novel Machine Learning (ML) based intrusion detection methods to automatically detect intruders globally and locally in VANETs. Compared to previous Intrusion Detection methods, our method is more robust to the environmental changes that are typical in VANETs, especially when intruders overtake senior units like RSUs and Cluster Heads (CHs). The experimental results show that this approach can outperform previous work significantly when vulnerable RSUs exist.
Zang and Yan [2021][11]proposed a Machine Learning-based Intrusion Detection System (IDS) for monitoring network traffic and detecting abnormal activities. This IDS framework integrates streaming engines for big data analytics, management and visualization. A Vehicular ad-hoc network (VANET) topology of multiple connected nodes with mobility capability is simulated in the Mininet-Wifi environment. Real-time data is collected using the sFlow technology and transmitted from the simulator to our proposed IDS framework. We have achieved high detection accuracy results by training the Random Forest as the classifier to label out the anomalous flows. Additionally, the network throughput has been evaluated and compared with and without deploying the proposed IDS. The results verify the system is a lightweight solution by bringing little burden to the network.
Bangui,et al [2021][12]proposed a hybrid ML model to enhance the performance of IDSs by dealing with the explosive growth in computing power and the need for detecting malicious incidents timely. The proposed approach mainly uses the advantages of Random Forest to detect known network intrusions. Besides, there is a post-detection phase to detect possible novel intruders by using the advantages of coresets and clustering algorithms. Our approach is evaluated over a very recent IDS dataset named CICIDS2017. The preliminary results show that the proposed hybrid model can increase the utility of IDSs.
Ercan et al [2021][13]proposed a Machine Learning (ML) mechanism that takes advantage of three new features, which are mainly related to the sender position, allowing to enhance the performances of IDS for position falsification attacks. Besides, it presents a comparison of two different ML methods for classification, i.e. k-Nearest Neighbor (kNN) and Random Forest (RF) that are used to detect malicious vehicles using these features. Finally, Ensemble Learning (EL) which combines different ML methods, in our case kNN and RF, is also carried out to improve the detection performance. An IDS is constructed allowing vehicles to detect misbehavior in a distributed way, while the detection mechanism is trained centrally. The results demonstrate that the proposed mechanism gives better results, in terms of classification performance indicators and computational time, than the best previous approaches on average.
Gonçalves et al [2021][14]proposed an Intelligent Hierarchical Intrusion Detection System (IDS) that divides the network into four levels and, each of them, into multiple clusters, enabling the usage of different Machine Learning (ML) based detection techniques. Thus, each level may use an algorithm that more suits its needs. The datasets used in this research work are publicly available and easily accessible, enabling the verification and comparison of the obtained results. However, these did not originate from real-world data but from simulation. The communications between all the hierarchy entities are secured using Vehicular Ad hoc Network Public Key Infrastructure and Attribute-Based Encryption with Identity Manager Hybrid (VPKIbrID). This hybrid model takes advantage of multiple techniques to fulfill several communication requisites for secure VANET communications. More precisely, the VPKIb-rID Attribute-Based Encryption (VPKIbrID-ABE) mode allows ciphering data to various targets without the need to cipher individually for each one of them.

Amaouche et al [2022][15]presented an optimized intrusion detection approach using realistic dataset called ToN-IoT derived from a large-scale heterogeneous IoT network, to achieve our model we used the mutual information technique for feature selection and the synthetic mi-nority oversampling technique (SMOTE) for class balancing Then to compare we tested various ML methods Logistic regression (LR), k-Nearest Neighbor (kNN), decision tree (DT), Random Forest (RF) and Support Vector Machine (SVM) for VANET Security.

Singh et al [2019][16]aimed to utilize the power of machine learning to detect wormhole attack in multi-hop communication of VANETs. Although various mechanisms have been proposed in the literature to detect this attack, the ML-based approach for wormhole has not been explored. To model the attack in VANET, we create a scenario of multi-hop communication using AODV routing protocol on NS3 simulator that uses the mobility traces generated by the SUMO traffic simulator. We run the simulation and collect the statistics generated using the flow monitor tool. These collected traces are preprocessed, and then k-NN and SVM are applied on this preprocessed file to make the model learn of wormhole attack. The performance of these two machine learning models is compared in terms of detection accuracy and four alarm types. Our study demonstrates that ML is a powerful tool, which can help deal with such attacks in a multi-hop communication of future generation CAVs.

3. Proposed methodology

This section discusses the proposed intrusion detection model in detail. Proposed model consist of four phases first one is preprocessing using min max normalization, second one is Modified Chicken Swarm Optimization (MCSO) based feature selection, third one is signature based intrusion detection using Improved Support Vector Machine (ISVM) fourth one is anomaly based intrusion detection based on Modified fuzzy c means clustering. Overall architecture of the proposed model is shown in figure 1.

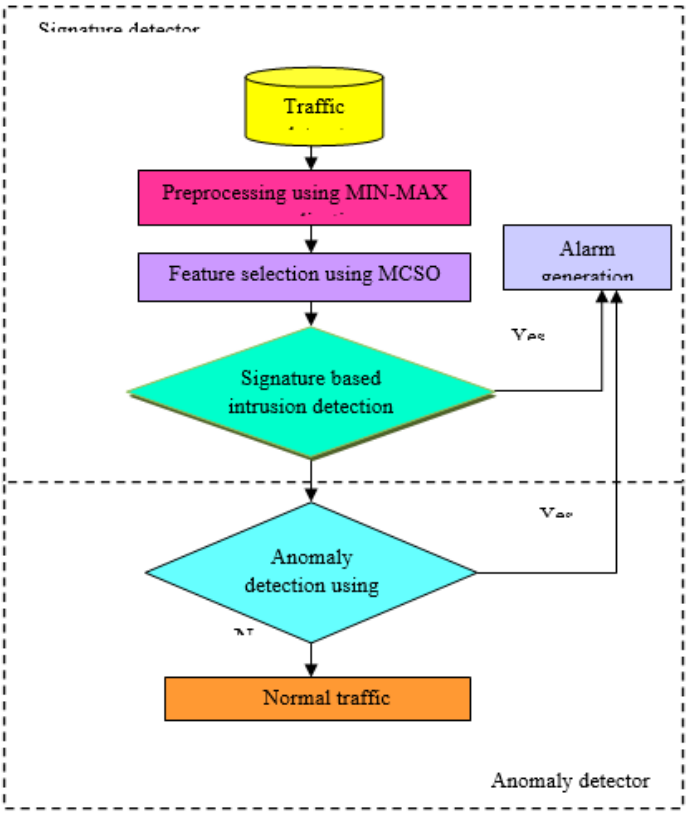

Figure 1.Overall architecture of the proposed model

3.1. Data normalization using min-max normalization

Need to normalize the data because input data might have scale variations which lead to provide inaccurate results to avoid this issues it is required to normalize the data [17]. This work uses Min-max Normalization model and the process of normalization entails converting numerical values into a new range using a mathematical function. Min-max normalization is one of the most common

ways to normalize data. The values in the dataset are normalized within the given range minimum and maximum value from dataset and each value are replaced according to the following formula (1).

$$v' = \frac{v - \min_A}{\max_A - \min_A}\left(\text{new\_max}_A - \text{new\_min}_A\right) + \text{new\_min}_A \quad (1)$$

Where,
A - Attribute data,
Min (A), Max (A) - minimum and maximum absolute value of A respectively
- New value of each entry in data
v - Old value of each entry in data
New_ max (A), new_ min (A) - max and min value of the range (i.e boundary value of range required) respectively.

3.2. Feature selection using Modified Chicken Swarm Optimization (MCSO)

After data normalization it sends for feature selection to reduce the time complexity and to increase the accuracy. This work using Modified Chicken Swarm Optimization (MCSO) for feature selection.

Chicken swarm optimization (CSO)

Chicken swarm optimization (CSO) is bio-inspire meta heuristic optimization algorithm[18].The algorithm mimics the hierarchal order of a chicken swarm and the behaviours of its individuals chickens. The hierarchal order of a chicken swarm is divided into several groups, each group consists of one rooster and many hens and chicks. Each type of chickens follows different laws of motions. A hierarchal order plays a significant role in the social lives of chickens[19,20]. The superior chickens in a flock will dominate the weak ones. There exist the more dominant hens that remain near to the head roosters as well as the more submissive hens and roosters who stand at the periphery of the group. Traditional CSO will easily falls into the trap of local optimal features.

To avoid this problem this work used mutation operator in CSO. This work used flip bit mutation. This mutation operator takes the chosen genome and inverts the bits. (i.e. if the genome bit is 1, it is changed to 0 and vice versa).

Mutation Chicken Swarm Optimization (MCSO)

The mathematical model of mutation chicken swarm optimization (MCSO) proposed in was based on the following rules that summarize the chickens' behaviours:

1)The chicken swarm is divided into several groups. In each groups there is a dominant rooster, following it some hens and chicks.

2) The fitness value of the chickens outlines the hierarchy of the swarm, the individuals with the best fitness will be the roosters each one will be a group leader, the individuals with the worst fitness values will be considered as chicks. The others would be the hens.

3) The swarm hierarchy, dominance relationship and mother-child relationship in a group will remain unchanged. These statuses only update every several (G) time steps.

4) The swarm consists of N virtual chickens divided as follow: RN, HN, CN, and MN which are the number of roosters, the hens, the chicks, and the mother hens, respectively. Each individual is represented by their positions in a D-dimensional space by

$$x_{i,j}\ (i \in [i,\ldots\ldots N], j \in [1,\ldots\ldots D]), \quad (2)$$

Rooster Movement: Roosters with better fitness values can search for food in a wider range of place than those with worse fitness values, such movement is depicted as in equations (8) and (9).

$$x_{i,j}^{t+1} = x_{i,j}^{t} * \left(1 + Randn(0, \sigma^2)\right) \quad (3)$$

$$\sigma^2 = \begin{cases} 1, & if\ f_i \le f_k\ , \\ exp\left(\frac{f_k - f_i}{|f_i| + \epsilon}\right) & otherwise \end{cases} \qquad k \in [1, N], k \ne i, \quad (4)$$

where xi;j is the selected rooster with index i, Rand n is a Gaussian distribution with mean 0 and standard deviation the smallest constant in the computer used to avoid zero-division-error, k a

randomly chosen roosters index selected from the roosters group, fi is the fitness value of the corresponding rooster xi.

Hen movement: Hens follow their group-mate roosters to search for food. Moreover, they would also randomly steal the good food found by other chickens, though they would be repressed by the other chickens. The more dominant hens would have advantage in competing for food than the more submissive ones. These phenomena can be formulated mathematically as in equations (6) and (7).

$$x_{i,j}^{t+1} = x_{i,j}^{t} + S1 * rRand * \left(x_{r_1,j}^{t} - x_{i,j}^{t}\right) + S2 * Rand * \left(x_{r_2,j}^{t} - x_{i,j}^{t}\right) \quad (5)$$

$$S1 = exp\left((f_i - f_{r1})/abs(f_i)+\epsilon\right) \quad (6)$$

$$S1 = exp\left((f_{r_2} - f_i)\right) \quad (7)$$

where Rand is a uniform random number over [0, 1]. r1 [1,….., N] is an index of the rooster, which is the ith hen's group-mate, while r2 [1,….., N, is randomly chosen index of a chicken (rooster or hen) from the swarm.

Chick movement: The chicks move around their mother to search for food. This is formulated as in equation (8).

$$x_{i,j}^{t+1} = x_{i,j}^{t} + FL * \left(x_{m,j}^{t} - x_{i,j}^{t}\right) \quad (8)$$

Where is the position of the ith chick's mother such that m [1;N], FL is parameter that represent how much speed a chick would follow its mother, to consider the differences between each chick FL is chosen randomly in the range [0, 2].

The feature space with each feature represented in an individual dimension and the span of each dimension ranges from 0 to 1 is very huge and hence requires an intelligent searching method to find optimal point in the search space that maximizes the given fitness function. The fitness function for the CSO is to maximize classification performance over the validation set given the training data, as shown in equation (9) while keeping minimum number of features selected.

$$f_\theta = \omega * E + (1 - \omega)\frac{\sum_i \theta_i}{N} \quad (9)$$

where is the fitness function given a vector with 0/1 elements representing unselected / selected features, N is the total number of features in the dataset, E is the classifier error rate and is a constant controlling the importance of classification performance to the number of features selected.

The used variables is the same as the number of features in the given dataset. All variable are limited in the range [0, 1], where the variable value approaches to 1; its corresponding feature is candidate to be selected in classification. In individual fitness calculation, the variable is threshold to decide the exact features to be evaluated as in the equation (10).

$$f_{i,j} = \begin{cases} 1 \; if \; X_{i,j} > 0.5 \\ 0 \; otherwise, \end{cases} \quad (10)$$

Where Xij is the dimension value for search agent i at dimension j. While updating the firefly position; solution, at some dimensions the updated value can violate the limiting constrains; [0, 1], and hence used simple truncation rule to ensure variable limits.

1. Initialize RN, HN, CN, MN, G;
2. Randomly initialize each chicken in the swarm
3. Xi (i = 1,2,……;N).;
4. Initialize the max numbers of iteration Tmax;
5. while T < Tmax do for each iteration
6. if T % G equals 0 then
7. Rank the chickens fitness values and establish a hierarchal order in the swarm;

8. Divide the swarm into different groups, and determine the relationship between the chicks and mother hens in a group;
9. end
10. for each chicken Xi in the swarm do
11. if Xi is a roster then
12. Update Xi's location using equation 4 ;
13. end
14. if Xi is a hen then
15. Update Xi's location using equation 6;
16. end
17. if Xi is a chick then
18. Update Xi's location using equation 9;
19. end
20. Evaluate the new solution using equation 10;
21. If the new solution is better than its previous one, update it;
22. end
23. end
24. Apply flip bit mutation to the updated solution
25. Evaluate the new solution using equation 10
26. end.

3.3. Signature based intrusion detection using Improved Support Vector Machine (ISVM)

Once get the feature subset from the feature selection phase need to classify those features to detect Signature based attacks in Vanet[21]. This work using Improved Support Vector Machine (ISVM) for signature based intrusion detection. Let us first consider, for simplicity, a supervised binary classification problem. Let us assume that the training set consists of N vectors (i = 1, 2, . . . ,N) from the d-dimensional feature space X. To each vector xi, associate a target . The linear SVM classification approach consists of looking for a separation between the two classes in X by means of an optimal hyper plane that maximizes the separating margin[22]. In the nonlinear case, which is the most commonly used as data are often linearly non separable, the two classes are first mapped with a kernel method in a higher dimensional feature space, i.e., (d′ > d).

However SVM using gaussian Radial Basis Function (RBF) kernel. The RBF kernel requires the computation of pair wise distances between all data points, which can be computationally expensive for large datasets. To overcome this problem this work using Improved Support Vector Machine (ISVM).

Improved Support Vector Machine (ISVM)

ISVM using polynomial kernel and it is expressed as

$$K(x_i, x) = [x_i . x + 1]^p \quad (11)$$

The membership decision rule is based on the function sign[f(x)], where f(x) represents the discriminant function associated with the hyper plane in the transformed space and is defined as

$$f(X) = \omega^* \Phi(x) + b^* \quad (12)$$

The optimal hyper plane defined by the weight vector , x is the kernel and the bias is the one that minimizes a cost function that expresses a combination of two criteria: margin maximization and error minimization. It is expressed as

$$\psi(\omega\Phi(x_i) + b) \geq 1 - \varepsilon_i, i = 1,2, \ldots, N \quad (13)$$ and

$$\varepsilon_i \geq 0, i = 1,2, \ldots, N \quad (14)$$

Where the is slack variables introduced to account for non separable data. The constant C represents a regularization parameter that allows controlling the shape of the discriminant function. In

this phase what are the features are classified as the normal traffic will send to the anomaly based intrusion detection phase.

3.4. Anomaly based intrusion detection based on Modified fuzzy c means clustering

In this phase anomaly based intrusion detection is performed on the features which are classified as a normal traffic in the signature based intrusion detection phase. In this work anomaly based intrusion detection is performed using Modified fuzzy c means clustering.

Fuzzy C Means Clustering (FCM)

A sort of clustering namely fuzzy clustering where every data point might be appropriate to more than one cluster [23, 24, 25]. Clustering or cluster analysis is nothing but data points allotment to clusters such that items in same cluster are as identical as probable, whereas items belonging to dissimilar clusters are as disparate as possible. Cluster identification is done through similarity measures which involve distance, connectivity, and intensity. The different similarity measures choice is dependent on data or application.

Disadvantages of FCM

In high dimensional signal scenario, certain features ought to be irrelevant and relevant however might possesses diverse significance in clustering. For improved clustering, it is necessary to include these features in the clustering method.

A Modified fuzzy c means clustering algorithm is suggested for mitigating these issues.

Modified Fuzzy c-Means Clustering

In this divergence amid traffic flows, SS (Signal Strengths), primary user access times, PDR (Packets Received/Packets Sent) contained by clusters computation can be done by weight function where weight allotment is done to every traffic flows, SS (Signal Strengths), primary user access times, PDR. Fuzzy c-means helps in data set X = {x1,…, xi,….xn} (1 i n)apportioning into c clusters based on membership degree matrix U=(uti)c*n when objective function J attains minimum value. The xi of X is p dimensional; uti denotes membership degree measures in which sample xi belongs to cluster center vt. Here, c clusters are marked by cluster centers V={v1,….vt,….vc} 1 t c , V is always set arbitrarily initially. Then membership degree uti computation is as follows:

$$u_{ti} = \frac{1}{\sum_{Z=1}^{c} (d_{ti}/d_{zi})^{2/(m-1)}} \quad (15)$$

Where dti denotes Euclidean distance amid sample xi to cluster center vt, m represents power

$$v_t = \frac{\sum_{i=1}^{n} u_{ti}^{m} d_{ti}}{\sum_{i=1}^{n} u_{ti}^{m}}$$

exponent. In iteration, cluster centres computation is as follows:

$$v_t = \frac{\sum_{i=1}^{n} u_{ti}^{m} d_{ti}}{\sum_{i=1}^{n} u_{ti}^{m}} \quad (16)$$

The objective function J is specified below

$$J = \sum_{i=1}^{n} \sum_{t=1}^{c} u_{ti}^{m} d_{ti}^{2} \quad (17)$$

The feature-weight learning is on the basis of weighted Euclidean distance. dij is frequently used Euclidean distance and denotes weighted Euclidean distance given below:

$$d_{ij}^{w} = \sqrt{\sum_{k=1}^{s} w_k (x_{jk} - v_{tk})^2} \quad (18)$$

Thus, objective function J specified in Eq. (19) will becomes:

Jw (U, v1, …. vc; X) =

$$J^{w}(U, v_1, \ldots v_c; X) = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^{m} (d_{ij}^{w})^2 \quad (19)$$

Then updated uij, wk and vik are obtained as follows:

$$u_{ij} = \frac{n(m\sum_{k=1}^{s} w_k(x_{jk} - v_{tk})^2)^{\frac{1}{m-1}}}{\sum_{h=1}^{c}\sum_{r=1}^{n}\left(m\sum_{k=1}^{s}(x_{jk} - v_{tk})^2\right)^{\frac{1}{m-1}}} \quad (20)$$

$$w_k = \frac{\left(\sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^{m}(x_{jk} - v_{tk})^2\right)^{-1}}{\sum_{h=1}^{s}\left(\sum_{i=1}^{c}\sum_{j=1}^{n}(x_{jk} - v_{tk})^2\right)^{-1}} \quad (21)$$

$$v_{ik} = \frac{\sum_{j=1}^{n} u_{ij}^{m} x_{jk}}{\sum_{j=1}^{n} u_{ij}^{m}} \quad (22)$$

The resultant M-FCM algorithm summary is as follows

Step 1: Fix maximum number of clusters c and a threshold value .Consider m be a proper constant.
Step 2: Initialize memberships and centers by FCM.
Step 3: Calculate wk based on Eq. (21).
Step 4: Calculate based on (20). Thus update the vik according to (22) by new computed .
Step 5: Calculate objective function J w by means of (19).
If it converges or difference amid two adjacent computed values of objective function J w is less than specified threshold then stop. Or else go to step 3.

4. Results and discussion

This section discusses the experimental results of the proposed model in detail. Proposed model is implemented in mat lab. Proposed IFCM is compared with the existing KNN and W-KMC models interms of preicison, accuracy, recall and f-measure. In this work, use the most recent dataset CICIDS2017 that contains the most up to date network attack scenarios, including the common type of attacks in VANET (like DoS Slowloris attacks and DDoS attacks). CICIDS2017 dataset covers the most cutting-edge frequent attack scenarios based on simulation of seven attack families, namely: brute force attack, heart-bleed attack, botnet, DoS attack, DDoS attack, web attack, and infiltration attack. A total number of 80 features were extracted based on the information present in the pcap file. The total number of records used in this experiment is 273 097. The dataset is divided into two parts using train-test_split, 80% for training and 20% for testing the model. Table 1.Shows the simulation parameters

Table 1. Simulation parameters

Parameter	Value
Simulation time	300 sec
Simulation Grid	1000 x 1000
Scenario	Two-way highway
Number of vehicles	100
Vehicle speed	15–45m/s
Data transfer rate	6, 12, 18, 27Mbps
Mac Layer	IEEE 802.11p
Packet size	100 bytes

Performance Metrics

1) Precision

Precision refers to the percentage of the results which are relevant and defined as

Precision=

$$\text{Precision}=\frac{\text{Truepositive}}{\text{truepositive}+\text{falsepositive}} \quad (23)$$

2) Recall

Recall refers to the percentage of total relevant results correctly classified by the proposed algorithm which is defined as

$$\text{Recall}=\frac{\text{Truepositive}}{\text{truepositive}+\text{FalseNegative}} \quad (24)$$

3) Accuracy

Accuracy is the fraction of predictions this model got right. Formally, accuracy has the following definition:

$$\text{Accuracy}=\frac{\text{Truepositive}+\text{TrueNegative}}{\text{Total}} \quad (25)$$

4) F measure

An F-score is the harmonic mean of a system's precision and recall values

2 x [(Precision x Recall) / (Precision + Recall)] (26)

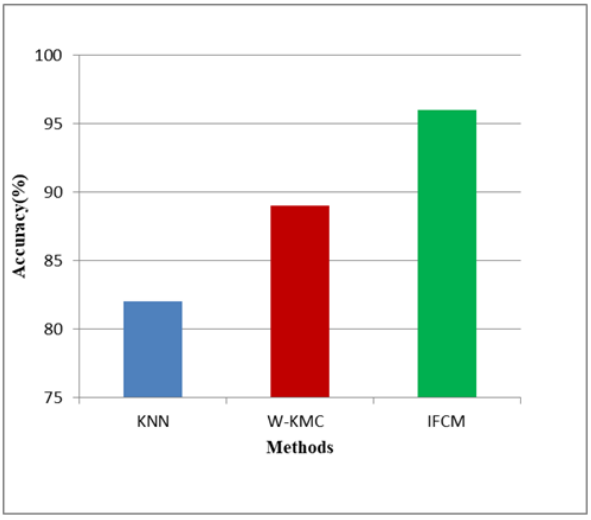

Figure 4.Accuracy results

Figure 4.Shows the accuracy performance metric comparison between existing KNN and W-KMC methods and proposed IFCM for intrusion detection. In the above figure X-axis represents the methods and the y-axis represents the accuracy results. This work using improved support vector machine and it increases the accuracy results. From the results it is concluded that the proposed IFCM model produces the higher accuracy results of 96% while the existing KNN and W-KMC models produces only 82% and 89% accordingly.

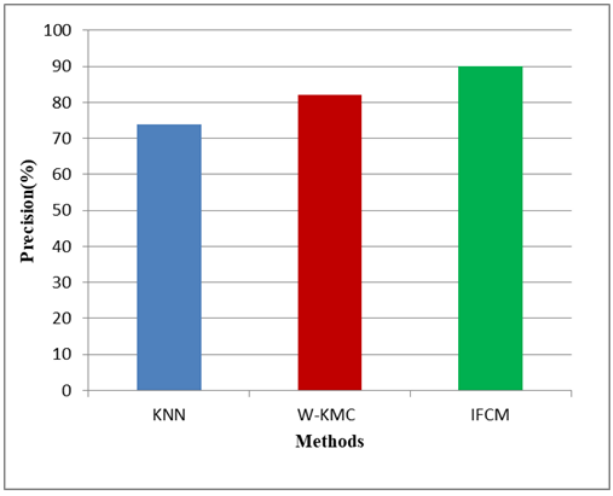

Figure 5.Precision results

Precision performance metric comparison between existing KNN and W-KMC methods and proposed IFCM for intrusion detection are shown in figure.5. In the above figure X-axis represents the methods and the y-axis represents the precision results. Proposed work using min-max for scale normalization and it increases the precision results. From the results it is concluded that the proposed IFCM model produces the higher precision results of 90% while the existing KNN and W-KMC models produces only 74% and 82% accordingly.

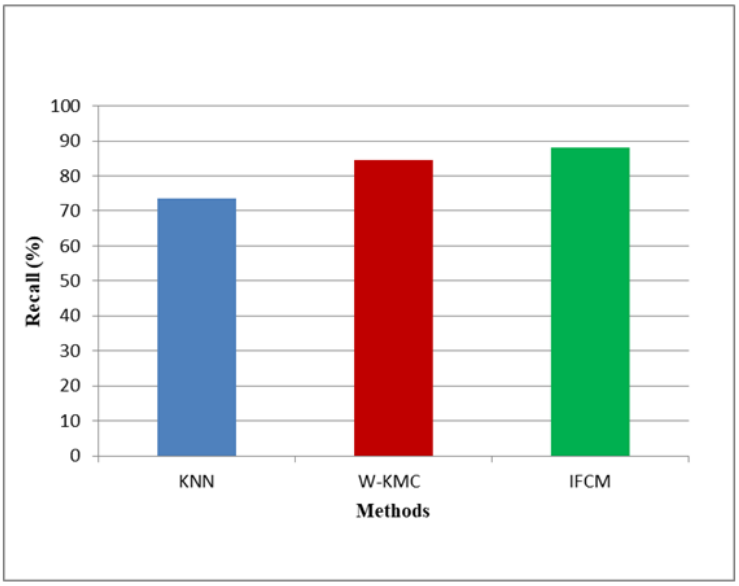

Figure 6.Recall results

Figure 6.Shows the performance comparison results for the existing KNN and W-KMC methods and proposed IFCM for intrusion detection interms of recall. In the above figure X-axis represents the methods and the y-axis represents the recall results. From the results it is concluded that the proposed IFCM model produces the higher recall results of 88% while the existing KNN and W-KMC models produces only 73.5% and 84.5% accordingly.

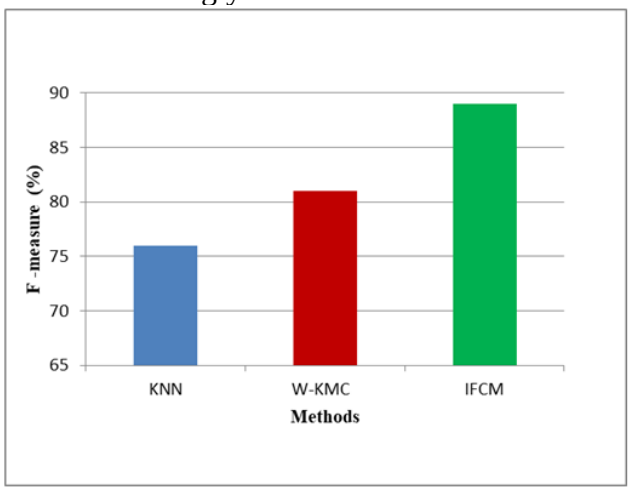

Figure 7.F -measure results

F-measure performance metric comparison between existing KNN and W-KMC methods and proposed IFCM for intrusion detection are shown in figure.7.Proposed model using Mutation Chicken Swarm Optimization which increases the f-measure results of the proposed model. In the above figure .X-axis represents the methods and the y-axis represents the f -measure results. From the results it is concluded that the proposed IFCM model produces the higher f -measure results of 89% while the existing KNN and W-KMC models produces only 76% and 81% accordingly.

5. Conclusion and future work

While Vehicular Ad-hoc Network (VANET) is developed to enable effective vehicle communication and traffic information exchange, VANET is also vulnerable to different security attacks. This work

aimed to provide an improved model for intrusion detection in Vanet. Preprocessing is done based on min max normalization method to normalize the input scale. Modified Chicken Swarm Optimization (MCSO) is applied for feature selection. Signature based intrusion detection is performed using Improved Support Vector Machine (ISVM).Output such as normal traffic from the Signature based intrusion detection phase will be used for anomaly based intrusion detection using Modified fuzzy c means clustering. Proposed model is evaluated interms of accuracy, recall, f-measure and precision. Results shows that the proposed model achieves the high accuracy than other existing models. Experimental results shows that the proposed model produces 96% accuracy and the other existing models such as KNN and W-KMC produces 82% and 89%.However support vector machine does not perform well with high volume data so need to use other models in future.

References

1. Sharma, S. and Kaul, A., 2018. A survey on Intrusion Detection Systems and Honeypot based proactive security mechanisms in VANETs and VANET Cloud. Vehicular communications, 12, pp.138-164.
2. Belenko, V., Krundyshev, V. and Kalinin, M., 2018, September. Synthetic datasets generation for intrusion detection in VANET. In Proceedings of the 11th international conference on security of information and networks (pp. 1-6).
3. Shu, J., Zhou, L., Zhang, W., Du, X. and Guizani, M., 2020. Collaborative intrusion detection for VANETs: A deep learning-based distributed SDN approach. IEEE Transactions on Intelligent Transportation Systems, 22(7), pp.4519-4530.
4. Karthiga, B., Durairaj, D., Nawaz, N., Venkatasamy, T.K., Ramasamy, G. and Hariharasudan, A., 2022. Intelligent intrusion detection system for VANET using machine learning and deep learning approaches. Wireless Communications and Mobile Computing, 2022.
5. Liang, J., Chen, J., Zhu, Y. and Yu, R., 2019. A novel Intrusion Detection System for Vehicular Ad Hoc Networks (VANETs) based on differences of traffic flow and position. Applied Soft Computing, 75, pp.712-727.
6. Alsarhan, A., Alauthman, M., Alshdaifat, E.A., Al-Ghuwairi, A.R. and Al-Dubai, A., 2021. Machine Learning-driven optimization for SVM-based intrusion detection system in vehicular ad hoc networks. Journal of Ambient Intelligence and Humanized Computing, pp.1-10.
7. Sedjelmaci, H. and Senouci, S.M., 2015. An accurate and efficient collaborative intrusion detection framework to secure vehicular networks. Computers & Electrical Engineering, 43, pp.33-47.
8. Ben Rabah, N. and Idoudi, H., 2022. A machine learning framework for intrusion detection in VANET communications. In Emerging trends in cybersecurity applications (pp. 209-227). Cham: Springer International Publishing.
9. Shams, E.A., Rizaner, A. and Ulusoy, A.H., 2018. Trust aware support vector machine intrusion detection and prevention system in vehicular ad hoc networks. Computers & Security, 78, pp.245-254.
10. Zeng, Y., Qiu, M., Ming, Z. and Liu, M., 2018. Senior2local: A machine learning based intrusion detection method for vanets. In Smart Computing and Communication: Third International Conference, SmartCom 2018, Tokyo, Japan, December 10–12, 2018, Proceedings 3 (pp. 417-426). Springer International Publishing.
11. Zang, M. and Yan, Y., 2021, April. Machine learning-based intrusion detection system for big data analytics in VANET. In 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring) (pp. 1-5). IEEE.
12. Bangui, H., Ge, M. and Buhnova, B., 2021. A hybrid data-driven model for intrusion detection in VANET. Procedia Computer Science, 184, pp.516-523.
13. Ercan, S., Ayaida, M. and Messai, N., 2021. Misbehavior detection for position falsification attacks in VANETs using machine learning. IEEE Access, 10, pp.1893-1904.
14. Gonçalves, F., Macedo, J. and Santos, A., 2021, October. Intelligent Hierarchical Intrusion Detection System for VANETs. In 2021 13th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT) (pp. 50-59). IEEE.

15. Amaouche, S., Benkirane, S., Guezzaz, A. and Azrour, M., 2022, November. A Proposed Machine Learning Model for Intrusion Detection in VANET. In The International Conference on Artificial Intelligence and Smart Environment (pp. 103-108). Cham: Springer International Publishing.
16. Singh, P.K., Gupta, R.R., Nandi, S.K. and Nandi, S., 2019. Machine learning based approach to detect wormhole attack in VANETs. In Web, Artificial Intelligence and Network Applications: Proceedings of the Workshops of the 33rd International Conference on Advanced Information Networking and Applications (WAINA-2019) 33 (pp. 651-661). Springer International Publishing.
17. Raju, V.G., Lakshmi, K.P., Jain, V.M., Kalidindi, A. and Padma, V., 2020, August. Study the influence of normalization/transformation process on the accuracy of supervised classification. In 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 729-735). IEEE.
18. Fouad, M.M., Hafez, A.I. and Hassanien, A.E., 2019. Optimizing topologies in wireless sensor networks: A comparative analysis between the Grey Wolves and the Chicken Swarm Optimization algorithms. Computer Networks, 163, p.106882.
19. Ali, B., Lashari, S.A., Sharif, W., Khan, A. and Ramli, D.A., 2021. An efficient learning weight of elman neural network with chicken swarm optimization algorithm. Procedia Computer Science, 192, pp.3060-3069.
20. Kumar, B.S., 2023. Application of chicken swarm optimization algorithm for multi objective scheduling problems in FMS. Materials Today: Proceedings, 72, pp.1457-1461.
21. Rodriguez-Galiano, V., Sanchez-Castillo, M., Chica-Olmo, M. and Chica-Rivas, M.J.O.G.R., 2015. Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines. Ore Geology Reviews, 71, pp.804-818.
22. Ebrahimi, M.A., Khoshtaghaza, M.H., Minaei, S. and Jamshidi, B., 2017. Vision-based pest detection based on SVM classification method. Computers and Electronics in Agriculture, 137, pp.52-58.
23. Askari, S., 2021. Fuzzy C-Means clustering algorithm for data with unequal cluster sizes and contaminated with noise and outliers: Review and development. Expert Systems with Applications, 165, p.113856.
24. Nida, N., Irtaza, A., Javed, A., Yousaf, M.H. and Mahmood, M.T., 2019. Melanoma lesion detection and segmentation using deep region based convolutional neural network and fuzzy C-means clustering. International journal of medical informatics, 124, pp.37-48.
25. Abdulshahed, A.M., Longstaff, A.P., Fletcher, S. and Myers, A., 2015. Thermal error modelling of machine tools based on ANFIS with fuzzy c-means clustering using a thermal imaging camera. Applied Mathematical Modelling, 39(7), pp.1837-1852.