Identification Of Malicious User To Combat Ssdf Using Trust Based Machine Learning Techniques In Cognitive Radio Networks

[1]Tephillah.S, [2]J.Martin Leo Manickam

[1,2]ECE,St.Joseph's Institute of Technology, Chennai,India

Abstract

Spectrum Sensing data falsification (SSDF) attack is one of the major threats in cognitive radio networks. In this paper a neural network (NN) machine learning (ML) is trained with multifactor trust based computed from the spectrum sensing output of the SU's to identifythe Malicious Users (MU's) in the SSDF attack.The performance of NN, Support Vector Machine (SVM), Naïve Bayes (NB) and Logistic Regression (LR) are compared. Also the performance of all the mentioned classifiers are validated with 'k'- fold cross validation method. Simulations have proved that the logistic regression ML is the best model to evaluate the trust based data set with 100% accuracy.

Key words: *SSDF,NN, SVM,NB,LR, trust based data.*

1.INTRODUCTION :

Spectrum scarcity is the major threat to the proliferation of wireless devices having an exponential increase in their usage day by day. Cognitive radio networks forms a solution to spectrum scarcity. It intelligently manages the spectrum usage by changingits transmission parameters according to the available spectrum and enables communication between users. Cognitive Radio (CR) provides unused spectrum to the secondary users- unlicensed user form the primary user –licensed user without causing interference to the latter when it is idle.

Cooperative spectrum sensing (CSS) exploits the spatial diversity for sensing inorder to improve the detection performance in case of multipath fading,shadowing and receiver uncertainty issues [1].Sensing techniques, hypothesis testing, data fusion, assignment of channel to user etc., forms the major elements of CSS. The two major fusion rules are soft decision and hard decision fusion. In the former the fusion center (FC) makes a global decision by fusing the sensing results of each SU using maximal ratio combining (MRC) or equal gain combining (EGC)[2]. In the latter, binary decision of each SUare combined in the FCusing fusion rules such as OR,AND,n out k etc.

Apart from the advantages of CSS, it is more liable to threatsthan distributed sensing. Spectrum data falsification attack (SSDF) forms the major threat in CSS as falsification of sensing report would hash the entire system. Literature provides many mitigation techniques for SSDF attack with trust and machine learning techniques.

2.RELATED WORKS:

In [3] feng.et.al have proposed a trust fluctuation clustering mechanism to correct the dynamic collusive SSDF attackers. Reputationbased CSS is proposed by Xinyu et al. [4] where the area under one FC is divided into different cells and users reputation is updated after each sensing time to identify the malicious user (MU)from legitimate users.W.Wang et al. having given a heuristic approach to iteratively identify MU's, by calculating suspicious level of user [5].

Recently machine learning (ML) based techniques have been employed to detect misbehaving nodes more accurately.Both supervised and unsupervised methodologies have been employed to detect the malicious users. The ML algorithm by means of training learns the features in the input data, and classifies the testing data depending on the learnt features. The authors in [65] have proposed a mitigation technique using k-means clustering to identify the MU's, by using the historical sensing data of the SU's. However, sensitivity in initializations of centroids would lead to different results.F.Farmani et al. have employed support vector data description (SVDD)classifier to exclude the MU's from the trusted nodes in decision phase at the FC. This method mitigates only always yes or always no attack as SVDD is one class classifier[6].In [66], Weighted Baseyain model has been proposed where the FC updates the weights of the SU's depending on their respective sensing results with reference to global decision. This method proves its proficiency unless the global decision does not go wrong.An unsupervised ML technique using Artificial Neural Network (ANN) have been proposed by [68].The method uses average suspicious value(ASD) to segregate the trusted SU's from MU's.The probability of miss-detection is high when noise is high forms the disadvantage.In [7] H.Zhu.et al. proposed a support vector machine learning scheme to mitigate different types of SSDF attacks. A classification accuracy index describing the behavior of SU is generated based on energy measurement values from multiple rounds of measurement. The simulations have proved best results even in case of low SNR and more number of MU's.

In this paper a mitigation method to identify the MU's using a trust based learn supervised neural network machine learning algorithm is proposed.

- Training efficiency of the ML relays on the input data-set. Thus the ML is trained with the multifactor trusts computed for different SU's using their sensing results measured over a period of time. These trusts acts as features for the training of the ML.
- A Learn supervised Neural network ML model is designed with analysis on the optimal choice of hidden layers, activation function, and epochs.
- Comparative analysis of NN,SVM,LR and NB with respect to precision, recall, f1 score and accuracy are performed to prove the best ML for the identification of MU's.

3. SYSTEM MODEL

The system model under consideration consists of one primary user(PU) whose spectrum is to be sensed for idle(free), 'S' secondary users who sense the PU's spectrum, one fusion center (FC) which finalizes the decision results and allocates the idle spectrum to SU. Let 'U' be the number of malicious users within the 'S' number of SU's. The MU's to be detected falls under three conditions (i) always 'yes' – the SU always reports that the spectrum is idle ,when it is not so. (ii) always 'No'- the SU always reports that the spectrum is busy even when it is not so. (iii) Mischievous – reports randomly irrespective of the true report.

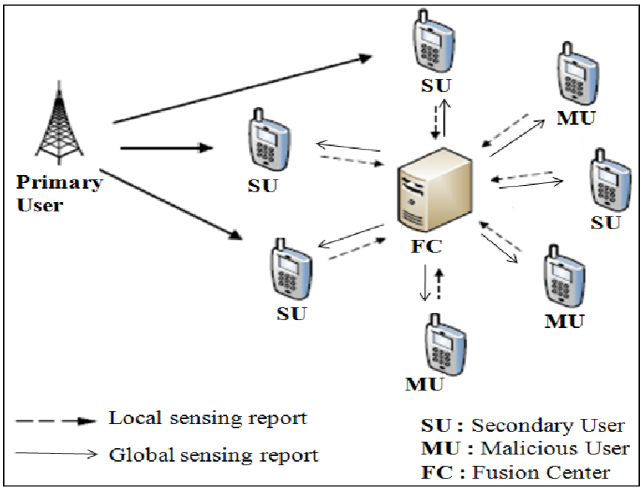

Fig 1: System Model

Usinga energy detection technique, each SU over a time slot 't' senses the spectrum and sends the sensed result to the FC. Each SU estimates the energy values under two hypothesis – H_0& H_1 as given by equation -1.

$$Y_k = \begin{cases} \sum_{u=1}^{m} n_k(u) ----> H_0 \\ \sum_{u=1}^{m} [h_k(u) + n_k(u)] \quad ^2 ----> H_1 \end{cases} (1)$$

Where, $n_k(u)$ is the additive white Gaussian noisewith zero mean & variance –σ_x .$.h_k(u)$is the uth sample of primary user signal at the kth SU [8]. H_0& H_1 are the hypothetical representation of absence and presence of PU signal. As the measured signal Y_k is the sum of squares of i.i.d Gaussian variable, using central limit theorem it can be shown that (for large values of 'k'),

$$Y_k = \begin{cases} \aleph\left(m\sigma_k^2, 2m\sigma_k^4\right)H_0 \\ \aleph\left((m+\delta_k)\sigma_k^2, 2(k+\delta_k)\sigma_k^4\right)\ H_1 \end{cases} \qquad (2)$$

Where, $\delta_k = \frac{\sum_{i=1}^{m} h_i^2}{\sigma_x^2} s^2(i)$

Eq(1)represents the measured signal by kthSU from set of {1,2, ... S} at a particular time 't_1'.

In order to obtain a final decision, the FC employs majority voting rule.The performance of the decision rule, is based on two metrics – Probability of detection (Pd) & probability of false alarm (pf). [9]

$$P_\sigma = Q_u\left(\sqrt{2\lambda}, \sqrt{\lambda}\right)(3) \qquad \mathrm{P_f} = \frac{\sqrt{\left(\mathrm{K}, \lambda/_2\right)}}{\sqrt{\lambda/_2}}$$

(4)

$$\mathrm{P_m} = 1 - \mathrm{P_f} \qquad (5)$$

Equation 5 gives the probability of miss detection.

4. NN MODEL:

Supervised and Unsupervised are the two types of ML algorithms. In supervised learning the model learns the features from the data set with which it is trained unlike unsupervised learning where the model learns from its environment. NN is one of the reliable ML which imitates the artificial functioning of the neuron. The NN model contains three layers :1. Input layer 2. Hidden layer 3.Output layer.Generally, the input layer contains the number of features in the data set as the number of neurons [11]. The abstract representation of the inputs is learnt by one or more hidden layers comprising of neurons. The output layer represents the binary classification with one neuron. The NN adapts to the changes in the inputs by adjusting its weights (w_i) and bias (b) in order to give the best possible output. Each neuron comprises an activation function (G) for learning complexities in the data. The output of each neuron can be mathematically modeled as the weighted sum of the input applied with the activation function as given in Eq (6)

$$O = G\left(\sum_{i=1}^{n} w_i\, x_i + b\right) \qquad (6)$$

Sigmoid, Rectified Linear Unit (ReLU), tanhare more commonly used activation functions. NN employ feedforward technique to calculate the output, that is the input fed to the input layer is processed by the hidden layers with its activation function and then the data is forwarded to the subsequent layers.

The optimal model selection of NN depends on the type of input dataset, number of hidden layers, number of neurons, activation function, optimization algorithm, epochs and batch size.The NN model consists of one input layer, one or many hidden layers and one output layer.

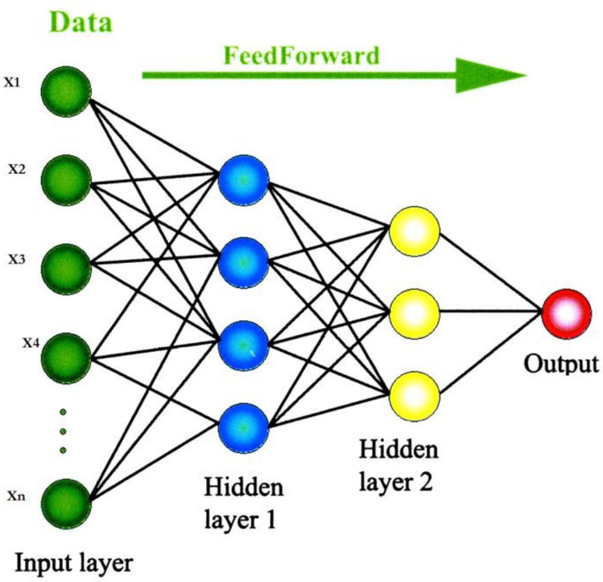

Fig.2 Model of NN

The selection of number of hidden layers and number of neuron for the proposed model was done based on the trial and error method as specified in [10] until the loss function reached minimum value and maximum accuracy. Also the number of epochs and batch size were determined based on the mentioned criteria. To overcome the vanishing problem ReLU activation function is used in the hidden layer [comparing ML] and sigmoid activation for the output layer. For optimization of the biases with high accuracy and speed – adam optimizer is used. And the Binary cross entropy is used as the loss function[13]. Alos to prevent overfitting of data K fold cross validation of data is peroformed.

Fig.2 shows the generalized model of NN, where [X1,X2…Xn] are the inputs with 'n' features. The features are the five different trusts calculated for the SU's. Mathematically, Eq.(7) represents the inputs. $X = [X_1, X_2, X_3, X_4, X_5\ ]$(7)

The output of the ith neuron of kth hiddenlayer can be formulated as Eq.(8)

$$h_i(k) = G_{ik}(\sum_{j=1}^{m} w_{ik} h_{i(k-1)} + b_{ik} \qquad (8)$$

Where, $h_i(k)$is the output, G_{ik} is the activation function of the kth hidden layer, w_{ik} and $b_{ik}$is the weight and bias of the ith neuron of kthhidden layer respectively.

$$Y_i(k) = G(\sum_{i=1}^{m} w_i\ (k) h_i(k))(9)$$

Eq.(9) shows the output of the NN model, where G is the activation function of the output layer and $\sum_{i=1}^{m} w_i\ (k) h_i(k)$ represents the summation of all the neurons output of kth the hidden layer. The update of weights and biases are done by gradient descent algorithm and the cross-entroy loss function gives the change in error when the weigths and bias are changed. The input data set to the NN model is generated from different trusts calculated from the sensing results of different SU's [12].The features are trust history, requite trust, registry trust, reliability trust and cumulative trust.

4.1 NAVIES BAYES

The Naïve Bayes ML is a family of ML algorithms uses Bayes theorem on the assumption that the features are independent. The high accuracy and speed of this algorithm on large data sets proves its advantage.The assumption of independency in the input features makes the algorithm to simplify its computations.The mathematical representation of Bayes Theorem by Eq.(10),

$$P\left(A/_B\right) = \frac{P(A)P\left(B/_A\right)}{P(B)}(10)$$

Where $P\left(A/_B\right)$ and $P\left(B/_A\right)$ are the conditional probabilities, P(A) and P(B) are the probabilities of the individual events. With regard to the proposed model , the input data set consists of various trusts computed using independent analytics, therefore Naïve Bayes ML could be used as training algorithm.

4.2 SVM

Support Vector Machine is supervised ML which creates a hyperplane (decision boundary) to discriminate the classes for the labeled data given as the input. It creates an optimal hyperplane with the help of support vectors. SVM also performs mathematical transformation on the input using Kernels. Sigmoid, Radial Basis functions, linear are few examples. The SVM is trained with input dataset and the performance is compared with the other ML's in this work.

4.3 LOGISTIC REGRESSION (LR)

LR is another type of ML form the field of statistics. It is fast and efficient classifier for binary classifications. The predicted output is transformed into logistic function using sigmoidal curve. The output of the LR can be used as the probability of input data belonging to the either class. The LR is also trained with the input data set for its performance comparison. The performance metrics chosen for comparison are accuracy, precision, recall and f1-score.

5.RESULTSand DISCUSSIONS

The dataset is generated for 70 different SU's, randomly distributed over an area of 500sqm. A single PU is considered to have a transmitting power of about -4db over this area. All the SU's perform CSS to the FC that computes the trust values based on the sensing results given by the SU's. The channel used for communication is assumed to be free from fading and shadowing. Nearly 293 categories for 70 SU's has been analyzed and 5 different trusts for each category has been computed. Thus the ML's are trained with 1465 samples (293*5).Also 80% of the input data set are used for training and 20% for validation.

a) Optimal Model of NN

First an optimal model of NN is chosen with one input layer with number of features as the number of neurons, three hidden layer with total of 32 neurons and one output layer with one neuron. The distribution of the neurons are considered as the 2^i,i=0,1,2..[Detection] until a highest value of accuracy is reached. Table.1 gives the comparative values for the number of neuron, number of hidden layers, epochs, batch size and optimizer for optimal model selection of NN.

Table:1 – Values for optimal NN model Selection

Sl.no	Activation Function	Number of Epochs	Batch Size	Number of Hidden Layers	Number of Neuron in Hidden Layer	Accuracy	Training Loss	Validation Loss
1.	Sigmoid	80	10	1	8	97.06	0.0881	0.2261
2.	Relu	100	10	2	16	98.13	0.0661	0.2011
3.	Relu	100	10	2	20	98.44	0.0578	0.1597
4.	Sigmoid	50	20	3	24	88.47	1.214	1.345
5.	Relu	100	10	3	24	95.99	0.0600	0.1442
6.	**Relu**	**100**	**10**	**3**	**32**	**99.47**	**0.0299**	**0.1432**

7.	Relu	100	10	2	36	95.72	0.0742	0.0742
8.	Relu	100	10	2	40	98.6	0.0482	0.1268

The optimum value was chosen at a point where accuracy was the highest with less value of training error and validation error. For the proposed system, with 32 number of neurons in the hidden layer – the model gave the highest accuracy of 99.47, training loss of 0.0299 and validation loss of 0.1432. For more than 40 neurons in the hidden layer the model accuracy was reduced by 0.87 but the training loss and validation loss was minimum. Also when the epoch was reduced below 100, there was a significant increase in both the training and validation loss. Thus the optimal model was selected based on the value of highest accuracy, minimum losses and 100 epochs.

Fig.3(a&b) gives the variation of training loss and validation loss with number of epochs for the NN model. As the number of epochs nears 100 the loss functions nears to minimum value.

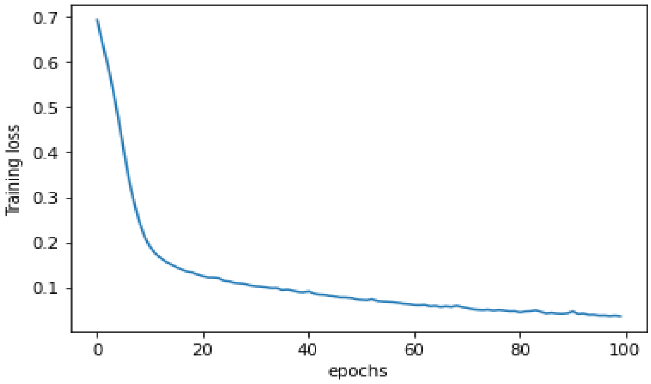

Fig.3(a) – Variation of Trainig Loss with epochs for NN model

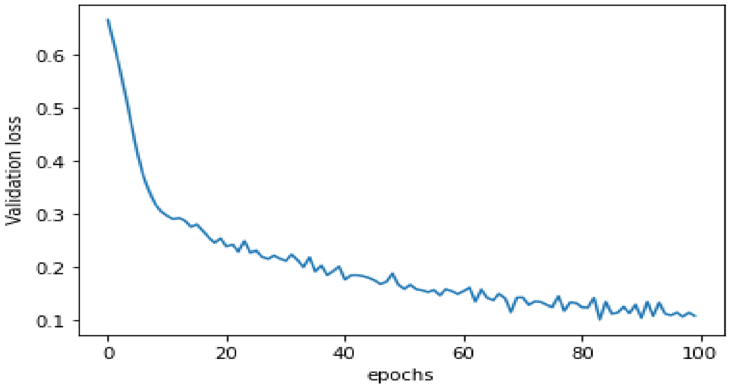

Fig.3(b) – Variation of Validation Loss with Epochs for NN Model

Fig.4 shows the variation of accuracy with the number of epochs. There is stipulated increase in the value of accuracy with number of epochs.

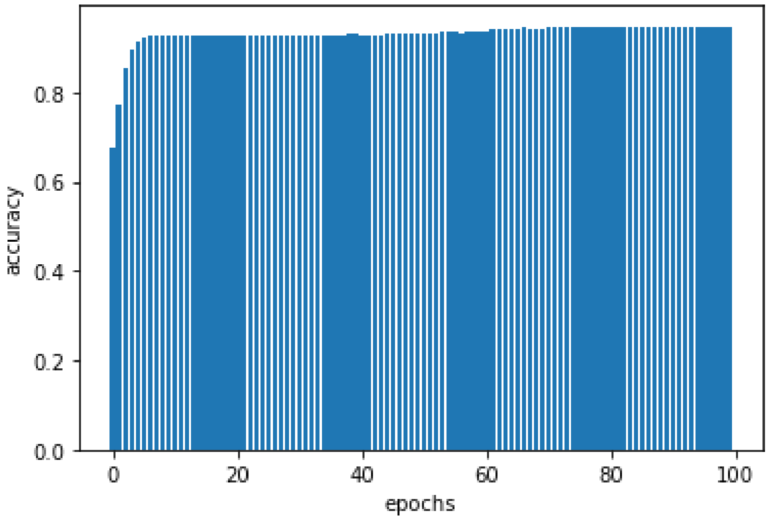

Fig.4- Variation fo Accuracy with epoch for the NN model

SVM, NB and LR ML's are trained with the same dataset and the performance metrics are measured. Table.2 gives the comparative values of various performance metrics for different ML's.

Table.2 : Performance Metrics of different ML's

SL.No	ML	Accuracy	Precision		Recall		F1-score	
			Malicious (Class 0)	Honest (Class 1)	Malicious (Class 0)	Honest (Class 1)	Malicious (Class 0)	Honest (Class 1)
1.	NN	99.47	95	75	98	50	96	60
2.	SVM	100	100	100	100	100	100	100
3.	NB	98	98	100	100	80	99	89
4.	LR	100	100	100	100	100	100	100

The Logistic Regression and the SVM proves its best in terms of accuracy, precision, recall and f1 score for the given data set.

b) Validation of the model

After the model is trained and tested, cross validation is performed using k-Fold cross validation (cv) to prove an unbiased evaluation of the optimal model fit on the training data set. Also cross validation will avoid overfittingof the data. In k-fold cv, the training set is divided into k smaller sets called as folds. Then the model is trained using k-1 folds as training data and the k^{th} fold is used for validation of the trained model. The performance metric for k fold is accuracy which is the average of the values computed using various k folds. Less variability of error, less bias, less computation are the advantages of k-fold cv.

The k-fold validation for different values of k, for all the ML's have been tabulated in Table.3.

Table.3 : 'k'-fold cross validation values for different ML's

Sl.No	ML	k-Fold	Accuracy (%)
1.	NN	2	96.07
		5	97.94
		10	97.79
		20	97.40
2.	SVM	2	92.8

		5	92.8
		10	92.8
		20	92.9
3.	NB	2	98.31
		5	98.31
		10	98.31
		20	95.6
4.	LR	2	100
		5	100
		10	100
		20	100

The k-fold cv is done for 4 different values of k, k=2,5,10&20 for all the ML’s. It can be inferred that the accuracy is 100% in the LR ML for all the values of ‘k’. LR has identified all the malicious users as malicious and honest SU’s as honest for the data used for cross validation. Thus LR ML is the best algorithm for the multifactor trust based data set.

6.CONCLUSION:

In this paper the performance metrics of different classifiers as NN, SVM, NB and LR are calculated for identifying the MU’s by training with a data set created by calculation of multifactor trusts for each SU. The multifactor trust data set relies on the spectrum sensing results of the SU’s. Also an optimal NN model was designed for the given data set and its performance was also measured. The ML’s were cross validated for different values of ‘k’ and the accuracy was measured. Overall LR ML proved to be the best classifier for the dataset. As future work, multiclass classifiers for identifying different types of SSDF attackers using the same and other ML techniques could be performed with the same dataset.

References:

[1] Olga León and K. P. Subbalakshmi (2017) “Cognitive Radio Network Security”,Handbook of Cognitive Radio, 30 pages.

[2] Doha Hamza,SoniaAissa and GhassaneAniba (2014) “Equal Gain Combining for Cooperative Spectrum Sensing in Cognitive Radio Networks”, IEEE Trans Wireless Communications 13(8), Pages 4334-4345.

[3]JingyuFeng ,Man Zhang ,Yun Xiao and HongzhouYue (2018) “Securing Cooperative Spectrum Sensing Against Collusive SSDF Attack using XOR Distance Analysis in Cognitive Radio Networks”, Sensors,18(2) pp(1–14).

[4]X. Wang, M. Jia, Q. Guo, X. Gu and G. Zhang (2017) "Reputation-based cooperative spectrum sensing algorithm for mobile cognitive radio networks," in China Communications, vol. 14, no. 1, pp. 124-134,

[5]L. Zhang, Q. Wu, G. Ding, S. Feng, J. Wang (2014),”Performance analysis of probabilistic softssdf attack in cooperative spectrum sensing”, EURASIP Journal of Advance Signal Processing (1) (2014) pp 1–9.

[6]Farmani. F,Jannatabad. A and Berangi. R (2011), “Detection of SSDF attack using SVDD algorithm in cognitive radio networks,” Proc. of Int. Conf. on Computational Intelligence, Communication Systems and Networks, pages 201-204.

[7] H. Zhu, T. Song, and J. Wu (2018) “Cooperative spectrum sensing algorithm based on support vector machine against SSDF attack,” in Proc. IEEE Int. Conf on Communication Workshops, Pages 1–6.

[8] H. Chen, M. Zhou, L. Xie, K. Wang and J. Li,(2016) ”Joint Spectrum Sensing and Resource Allocation Scheme in Cognitive Radio Networks with Spectrum Sensing Data Falsification Attack,” *IEEE Trans. on Vehicular Tech.*, vol.65, no.11, pp.9181 9191.

[9] Paschalis C. Sofotasios, MikkoValkam, Theodoros A. Tsiftsis, Yury A. Brychkov, Steven Freear and George K. Karagiannidis (2014), "Analytic Solutions to a Marcum Q−Function-Based Integral and application in Energy Detection of Unknown Signals over Multipath Fading Channels", IEEE Trans. Inf. Theory,60(12) pp(7798−7823).

[10] R. Devi, B. S. Rani, and V. Prakash, "Role of hidden neurons in an elman recurrent neural network in classification of cavitation signals," *International Journal of Computer Applications*, vol. 37, no. 7, pp. 9–13, 2012.

[11]"An Artificial Neural Network Approach for Detecting Spectrum Sensing Data Falsification Attacks." ukdiss.com. 11 2018.All Answers Ltd. 12 2020 <https://ukdiss.com/examples/artificial-neural-network-data-falsification-attacks.php?vref=1>.

[12] S. Tephillah, J. Martin Leo Manickam, "An SETM Algorithm for Combating SSDF Attack in Cognitive Radio Networks", Wireless Communications and Mobile Computing, vol. 2020, Article ID 9047809, 9 pages, 2020. https://doi.org/10.1155/2020/9047809.

[13]Sarmah, R., Taggu, A. &Marchang, N. Detecting Byzantine attack in cognitive radio networks using machine learning. Wireless Netw 26, 5939–5950 (2017). https://doi.org/10.1007/s11276-020-02398-w.