2023 IEEE 9th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)

2023 IEEE 9th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS) | 979-8-3503-1293-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/BigDataSecurity-HPSC-IDS58521.2023.00023

Malicious ADS-B data Generation Based on Improved GAN

Jin Lei
School of Safety Science and Engineering
Civil Aviation University of China
Tianjin, China
jlei@cauc.edu.cn

Ruifang Jiang
School of Safety Science and Engineering
Civil Aviation University of China
Tianjin, China
2020071025@cauc.edu.cn

Zhijun Wu
School of Safety Science and Engineering
Civil Aviation University of China
Tianjin, China
zjwu@cauc.edu.cn

***Abstract*—Automatic dependent surveillance broadcast (ADS-B) system sends messages over unencrypted wireless channels without any information integrity protection measures, and its messages are at risk of interception and tampering, which can easily lead to impersonation and forgery attacks. At present, although the ADS-B data anomaly detection model based on machine learning has excellent performance in predicting normal samples, the machine learning model may face different degrees of risk in each stage of its life cycle due to the existence of a large number of attackers in real scenes. To build secure and reliable machine learning systems, exploit potential vulnerabilities. Aiming at the ADS-B abnormal data detection model based on machine learning, this paper studies a construction method of poisoning data with strong applicability and establishes attack model. By injecting malicious data generated by the Generative adversarial network into the machine learning model, the performance of the trained model deteriorates and data misclassification occurs. Experimental results show that the malicious ADS-B data generation method proposed in this paper achieves good results, which lays a foundation for optimizing system defense technology and guaranteeing ADS-B security.**

Keywords*—ADS-B, Generative adversarial network, machine learning, malicious data, security

I. INTRODUCTION

ADS-B system is a new air traffic control surveillance technology based on satellite positioning for traffic monitoring and information transmission. The system does not need manual operation. It can automatically retrieve the position, speed, aircraft identification number and other parameters [1,2] from relevant aircraft onboard equipment to provide accurate aircraft position information for the aircrew. As a result, most commercial aircraft in the world are equipped with ADS-B by now.

However, as ADS-B comes into use, its vulnerabilities are causing concern. ADS-B system continuously broadcasts information through unencrypted wireless channels without any information integrity protection measures, and is vulnerable to attacks. In recent years, ADS-B attackers have analyzed and modeled various attack types of ADS-B data and achieved good results. For example, Costin et al. [3] proved that attackers can transmit spoofed data to ADS-B receivers through cheap radio, which shows the simplicity and feasibility of attack implementation. Therefore, it is urgent to solve the security risks of ADS-B system.

With the increasing availability of publicly accessible ADS-B data, the use of machine learning methods to enhance aviation safety is becoming more common. Although machine learning detection method improves ADS-B machine learning detection method improves ADS-B abnormal data monitoring, it faces many security threats in different life cycles. For ADS-B anomaly data detection model based on machine learning, this paper proposes an adversarial sample generation method, and the main innovative research work is as follows.

monitoring, it faces many security threats in different life cycles. For ADS-B anomaly data detection model based on machine learning, this paper proposes an adversarial sample generation method, and the main innovative research work is as follows.

In this paper, we investigate a more adaptive method of constructing malicious data for ADS-B anomaly data detection model, that is, the generative adversarial network generates adversarial samples, in which the Generator is responsible for generating perturbations to the data, and the Discriminator is responsible for ensuring that the generated adversarial samples are real. By improving the traditional GAN, an ADS-B attack sample generation is proposed, which cause the model to give a false output with high by adding malicious ADS-B data to normal sample [4]. This attack method is called "adversarial attack".

The article is organized as follows. Section 2 discusses the application of machine learning algorithm in ADS-B anomaly detection and the threats faced in recent years, and then describes the structure of the GAN in Section 3, focusing on the improvement of the GAN and the generation of malicious data. Section 4 introduces the experimental process and analyzes the experimental results, and concludes the research of this paper in Section 5.

II. RELATED WORK

In recent years, ADS-B data has been used to train machine learning models to detect ADS-B attacks, and machine learning-based defense techniques are easy to integrate with ADS-B systems because they do not require infrastructure changes or additional sensors. In 2018, Habler et al [5] trained LSTM encoder-decoder model to detect common airborne attacks. In 2019, Akerman et al. [6] have extended the same model using a convolutional neural network to analyze all aircraft within a specific area and represented them as sequential image streams. Each flight analyzed by these methods has large-scale historical data in order to detect abnormal trajectories [7].

Previous work has demonstrated that in a specific flight path training the recursive neural network can real-time detect the ADS-B attack. Although machine learning detection means to improve the abnormal data monitoring, it is confronted with many security threats, for example, attackers may inject malicious data into model input samples or steal model parameters, thus compromising the confidentiality, availability and integrity of the model [8].

National Key R&D Program of China (2022YFB3904503), the Natural Science Foundation of Tianjin China (21JCZDJC00830), the Fundamental Research Funds for the Central Universities of China (ZXH2012P004).

979-8-3503-1293-5/23/$31.00 ©2023 IEEE

DOI 10.1109/BigDataSecurity-HPSC-IDS58521.2023.00023

In 2013, Szegedy et al. [9] first introduced the concept of "adversarial examples" and proposed the L-BFGS method, to makes neural network misclassification by finding the minimum loss function additive term, which has a high success rate of attacks but also has a high computational cost. To reduce the computational cost, Goodfellow et al. [10] proposed the Fast Gradient Notation Method (FGSM) in 2014. The method is used to perturb the inputs to the model before back propagation, which is an early form of adversarial training. In 2018, Xiao et al. [11] proposed an adversarial sample generation method AdvGAN based on GAN, where a generator is used to generate adversarial interference, a discriminator is used to identify the adversarial sample. This method has a high success rate and is not easy to detect.

Since the input to a machine learning algorithm is in the form of Numeric vectors, an attacker designs a specific Numeric vector to make the machine learning model misclassify, which is called an adversarial attack. Although the effect of adversarial attack on the classifier is less than that of noise jamming, the probability of being misclassified by the classifier is much higher than that of noise jamming, which means that adversarial samples will become a blind spot of machine learning training algorithms [12].

In this paper, an attack is launched on a machine learning model with known algorithms and parameters. An improved GAN is used to generate adversarial samples similar to the original ADS-B samples, and by adding malicious ADS-B data to the normal samples, it achieves the goal of making the machine learning model make a wrong judgment on the carefully constructed malicious data. The improved GAN uses conditional generator and sampling training method to generate samples similar to the original sample distribution [13]. Experimental results show that the adversarial sample using the improved GAN has a high success rate of attack under the condition of defense.

III. Malicious ADS-B Data Generation Using Improved GAN

Generative adversarial network is composed of two networks, these two networks have adversarial goals in their training. The discriminator tries to maximize its classification accuracy, while the goal of the generator is to fool the discriminator [14]. The generative network has to continuously optimize the generated synthetic data to make the discriminative network unable to discriminate the truth of the data, and the discriminative network continuously optimizes to make the judgment result more accurate. The relationship between the two forms an adversarial, hence the name adversarial network. [15]. At the end of the learning, the generator is able to produce a composite data distribution similar to the real data. The basic architecture of the GAN is shown in Fig 1.

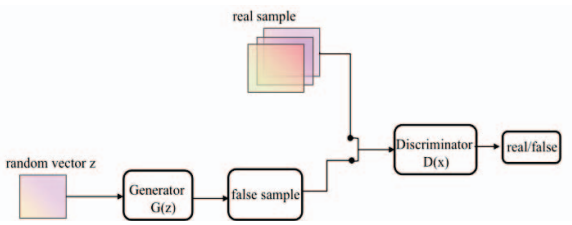


Fig. 1. Basic architecture of a GAN network.

Optimization principle, both generative and discriminative networks contain loss functions. Back Propagation and optimization methods are used to achieve parameter tuning, and constantly enhance the performance of generative and discriminative network, and the finally state of maturity of the generative and discriminative network is the learning of reasonable mapping functions [16].

GANs are very flexible in representing data, GANs and their many extension models are increasingly used in a wide range of applications, such as EhrGAN generation to generate enhanced medical records, MedGAN to combine autoencoders and GAN to generate heterogeneous non-time series , etc. [17].

A. Improved GAN

Due to the tabular form of ADS-B data obtained from website, several features of tabular data present challenges for GANs.

Mixed data types: Tabular data contains many types. If both discrete and continuous columns are to be generated, the GAN network must use both Softmax and Tanh for output.

Non-Gaussian distribution: Continuous values in the tabular data follow the non-Gaussian distribution, where the min-max transformation method will lead to the problem of vanishing gradients.

One-hot encoding vector: When generating synthetic data, Softmax is used to train the generation models to generate the probability distribution of each mode, and the representation of the real data by the one-hot vector. That's inappropriate for the discriminative network only checks the distribution sparsity without considering the authenticity of the whole sample.

Highly unbalanced categorical columns: In the dataset, most categorical columns are highly unbalanced. This can cause severe pattern collapse and lead to inadequate training of small categories [18].

In order to solve the above problems, normalization for patterns was invented in CTGAN to overcome non-Gaussian distributions, designed a condition generator trained by sampling to deal with unbalanced discrete columns.

1) Normalization for patterns: CTGAN devised a method for schema normalization to handle columns with complex distributions. There is mode-specific normalization for continuous columns, which converts continuous values of arbitrary range and distribution into bounded vector representation suitable for neural networks [19].

*2) Conditional generator and sampling training method:*A conditional generator is introduced to solve class imbalance problem of discrete columns. Integrating conditional generators into GAN's architecture requires solving the following problems.

a) Designing the representation of conditions.

b) The generated rows need to preserve the given conditions.

c) Conditional generators learn conditional distribution of real data: The output of a conditional generator is a one-hot conditional vector, denoted as cond.

Using the Variational Gaussian mixture Model (VGM), the original distribution can be fitted by the following formula.

$$p(c) = \sum_{k=1}^{k} \pi_k N(c \mid \mu_k, \sum_k)^3 \quad (1)$$

where, c is the value of a continuous column in Formula (1), and μ_k is the weight of the mode. Notice that the actual column is treated as a variable, and the covariance matrix degrades to variance σ^2 . For a well-fitted VGM, the probability ρ_k from any value c of each model can be calculated. If a pattern k^* is sampled, the pattern is regularized.

$$\alpha = \frac{c - \mu_{k^*}}{4\sigma_{k^*}} \quad (2)$$

where, α is a scalar and β represents the one-hot vector whose k^* mode is sampled. Note that in most case, c follows the selected Gaussian distribution, so this regularization is reasonable. For discrete columns, one-hot vectors are used.

The final row of data can be tabulated as follows.

$$r = \alpha_1 \oplus \beta_1 \oplus L \ \oplus \alpha_{N_c} \oplus \beta_{N_c} \oplus d_1 \oplus L \ \oplus d_{N_d} \quad (3)$$

Formula (3) is the final output result of the conditional generator after fair sampling, where d is the one-hot encoding representation of discrete columns.

3) Network framework of CTGAN: Because the columns in a row have no local structure, both generators and discriminators use two fully connected hidden layers to capture all possible associations between the columns.

Batch normalization and RELU activation functions are used in the generator[20]. Tanh generates scalar values α_i and Gumbel Softmax generates pattern indicators β_i and discrete values d_i .

In the discriminator, we use the Leaky relu function and Dropout method on each hidden layer. Generate a random vector z of constant length, so that each component is independently sampled from N(0,1). The conditional generator is used to sample the vector $cond$, whose length is the sum of the categories of each discrete column, and the structure of CTGAN is shown in Fig 2 .

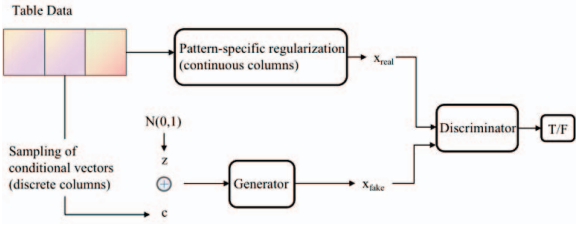


Fig. 2. Basic architecture of a CTGAN

B. Evaluation Indicators

In this paper, the accuracy, Recall, Detection and F1-score values of the anomaly detection model before and after the injection of adversarial samples were compared to determine the effectiveness of the attack, which are calculated by the confusion matrix.

TABLE I. CONFUSION MATRIX.

	Predict a positive	**Predict a negative**
The label is positive	True Positive（TP）	False Negative（FN）
The label is negative	False Positive（FP）	True Negative（TN）

$$\text{precision} = \frac{TP}{TP + FP} \quad (4)$$

$$F1 = \frac{2 * \text{precision} * \text{Recall}}{\text{precision} + \text{Recall}} \quad (5)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (6)$$

$$\text{Detection} = \frac{TN}{FP + TN} \quad (7)$$

IV. EXPERIMENT AND RESULT ANALYSIS

The experimental environment of this thesis is Intel(R) Core(TM) I5-5257U processing frequency is 2.70GHz, Python version is python3.8, and the weight decay value of the generator of Adam optimizer is set to 0.00001. Adam optimizer sets the discriminator weight decay value to 0.00001, Batch_size=100, number of iterations is 50, and sample num_rows is set to 40. From https://flightadsb.variflight.com/tracker/123.75497 8, 34.885186/3 obtain a total of 17000 ADS-B data of 25 different flights as training samples in the form of tabular data. The ADS-B data of each aircraft is between 600-1000, including time, altitude, speed, longitude, angle, etc.

The experiment includes simulation of interference data, anomaly detection and adversarial sample generation.

A. Analog interference data

The generation method of spoofing data is as follows:

1) Random noise: The 150 ADS-B height and speed flight data in the middle are multiplied by a random floating-point number between 0 and 2.

2) Route replacement: The middle 300 data points are replaced with another flight data point with a similar flight path (e.g. climbing aircraft), and the rest data points are not processed.

3) Height offset: The height features of the middle 300 ADS-B data will add random numbers in the interval of -500,500, and the rest data will not be processed.

4) Speed offset： The middle 400 data are incrementally changed in multiples of 5 with respect to the speed characteristic information of ADS-B data. Specifically, the speed characteristic contained in the 101-200 ADS-B data is increased by 5, and the 201-300 data is increased by 10, and so on. The rest of the data is not processed.

5) Replay attack: 91-100 pieces of ADS-B trajectory data are continuously sent 10 times consecutively, and the receiving end receives the 91-100 pieces of track that have been delayed for 10 times, and the remaining data are not processed. The normal flight path is compared to the flight path after the attack as shown in the figure below.

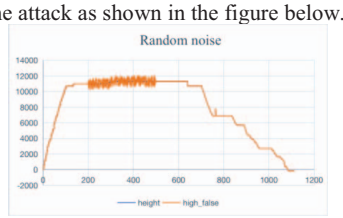


Fig. 3. Random noise.

Fig. 4. Route replacement.

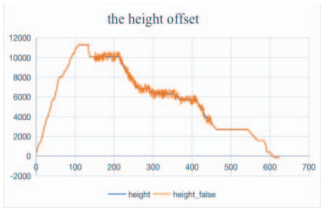


Fig. 5. Height offset.

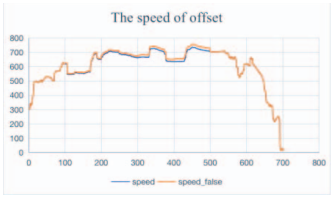


Fig. 6. Speed offset.

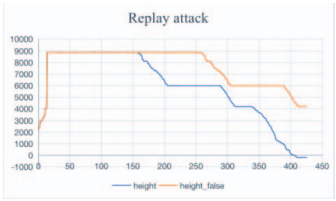


Fig. 7. Replay attack.

B. *Anomaly detection*

The data is preprocessed before anomaly detection, that is, missing value processing and data normalization.

Missing value processing: Data with obvious abnormalies are processed using regular expressions. For data with missing values, if a certain data has multiple missing values of attributes, the tuple is directly deleted. For missing data, the average value is used to supplement the data.

Normalization: Due to the difference in the number of feature levels of the data, in order to eliminate the adverse effects caused by odd sample data, in this paper, the zoom range value of each feature, normalized to [0, 1], for the characteristics of each dimension D_i.

$$D = \frac{D_i - D_{min}}{D_{max} - D_{min}} \tag{8}$$

In this paper, LSTM, Seq2seq model, GRU and BiGRU are selected to detect anomalies in the above five ADS-B attacks. They have similar network structures, as shown in Fig 8.

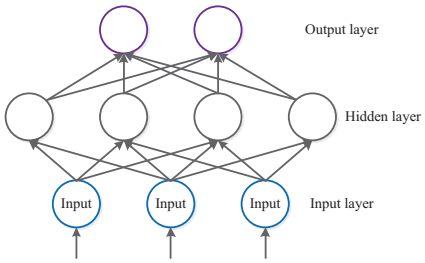


Fig. 8. Network structure diagram.

The threshold value of anomaly detection is determined by referring to the method of literature[20], then the threshold is defined as the decision boundary which is transformed into a classification problem. In the classification problem, it is assumed that the function is actually a composite function, and the inner function is a linear regression function with an activation function $g(z)$ on the outside.

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1+e^{-\theta^T x}} \tag{9}$$

$$g(z) = \frac{1}{1+e^{-z}} \tag{10}$$

The value of $g(z)$ is between [0,1], in fact, the output can be regarded as probability, A threshold can be defined. Therefore, judgment classification can be transformed into the relationship between the output value of $g(z)$ and a certain threshold value, and the approximate value of z can be obtained from the curve of $g(z)$.

The threshold Y of the abnormal score is determined by calculating the abnormal score of the training set, and the threshold was defined as 95%. Data greater than 95% above the decision boundary is regarded as abnormal data, whereas data below the decision boundary is regarded as normal data.

A total of 300 pieces of real data and spoofing data are randomly selected, which were divided into two categories according to the above method, namely, real data and fake data. As shown in Fig 9 (blue is normal data, red is abnormal data, auxiliary line is Y=95%).

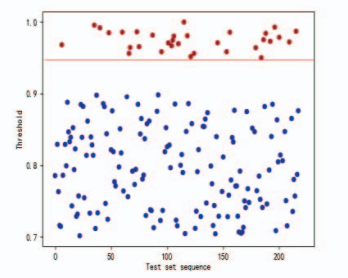
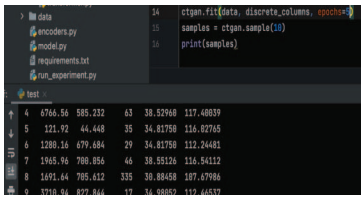


Fig. 9. Anomaly detection classification diagram.

C. Generation of adversarial samples

Read data into data frames and eliminate unnecessary columns from tabular data. We pass to the model, The CTGAN model learns these data and then samples the composite data to see how well the model captures the above features. Then will produce a table identical to the model fit table and populated with new data similar to the original data. When epoch=5, the longitudinal and latitudinal deviation of the generated data is large and the authenticity is low. When epoch=50, the longitudinal and latitudinal degrees of the generated data are found to be stable. Fig.10 shows the comparison of generated data with different epoch values.



(a) epoch=5

(b) epoch=50

Fig. 10. Generating a data comparison diagram.

D. Analysis of experimental results

After training, CTGAN understands the distribution of training data to generate a distribution similar to that of the training data. We compared the real flight trajectory with the flight trajectory of the generated data. It can be seen from Fig 11 that the distribution of the raw data is very similar to that of the synthetic data. The top image corresponds to the distribution of real data, and the bottom image corresponds to the distribution of synthetic data.

(a)

(b)

Fig. 11. Comparison of flight paths.

When the adversarial samples generated by the attack model are added to the machine learning training dataset for anomaly detection, it can be found that the added synthetic data is below the decision boundary, that is, the data is identified as true, as shown in Fig 12. (Blue is real data, red is spoofing data, light blue is adversarial sample).

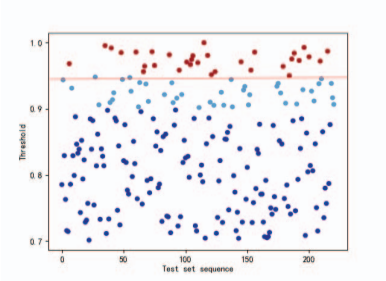


Fig. 12. Anomaly detection classification diagram of adversarial samples.

The accuracy rate can reflect the distance between the generated samples to the real data stream shape, and a high accuracy rate means that the generated samples are close to the data stream shape. In this paper, we evaluate the performance of CTGAN by calculating ac_base, ac_train, and ac_test for comparative analysis.

The training sample set, validation set and sample set generated by CTGAN were defined as Dt, Dv and Dg, respectively. Then the classifier was trained on Dt and the accuracy rate was calculated on Dv, and the accuracy rate was denoted as ac_base. The classifier is trained on Dg and the accuracy is calculated on Dv, which is denoted as ac_train. The classifier is trained on the training set Dt and the accuracy is calculated on Dg, which is denoted as ac_test.

Compare ac_base and ac_train. When ac_train is smaller than ac_base, it indicates that CTGAN has problems. If ac_train is close to ac_base, CTGAN generates high data quality. Compare ac_base and ac_test. Ideally, the values are close. If ac_test is very high, then CTGAN is overfitting. If ac_test is low, the synthesized data generated by CTGAN is of poor quality. Table 2 shows that the values of ac_base,ac_train and ac_test are close to each other.

TABLE II. CTGAN EVALUATION METRICS.

The evaluation index	ac_base	ac_train
accuracy%	84.31	85.07

Statistical measures used for serial similarity measures, such as sum square error (SSE), root mean square error (RMSE), mean relative error (MRE), and mean absolute error (MAE), which are used to measure synthetic data stability. A perfect model is equal to 0 when the predicted value exactly matches the true value; the larger the error, the larger the value. Table 3 shows that all four similarity

measures are close to 0. Therefore, the error between the synthetic data of CTGAN model and the real data is small.

TABLE III. SYNTHETIC DATA EVALUATION METRICS.

The evaluation index	MAE	MSE	RMSE	SSE
CTGAN	0.0089	0.0091	0.0099	0.0092

In this paper, Seq2seq, LSTM, GRU and BIGRU are selected to detect anomalies in ADS-B data of five kinds of spoofing interference. Table 4 shows the experimental results of detection rate, recall rate and accuracy rate of different anomaly detection methods under five forms of attack. It is observed that the detection rate and accuracy rate of the abnormal mode of route replacement are lower than other abnormal modes, because the length of the abnormal sequence segment of route replacement is shorter than the length of the whole test sequence segment and the change is not obvious. The deviation of height offset injection is small, leading to no obvious height change, so height offset is also difficult to detect. The detection rates of other attacks were all above 93%.

TABLE IV. ACCURACY, F1 SCORE AND DETECTION RATE OF EACH METHOD.

methods	The evaluation index	Random noise	Route replace	The height offset	The speed offset	Replay attack
Seq2seq%	*Accuracy*	93.16	92.31	93.85	95.32	93.67
	F1 score	95.46	95.22	84.79	89.60	92.35
	detection	93.23	87.91	92.56	96.89	96.77
LSTM%	*Accuracy*	91.13	88.62	91.35	92.05	90.87
	F1 score	94.12	93.09	78.18	81.74	89.26
	detection	90.39	87.23	91.87	94.73	95.81
GRU%	*Accuracy*	90.02	88.03	90.17	91.23	90.15
	F1 score	93.30	92.87	77.34	81.89	86.70
	detection	90.02	86.27	90.73	93.54	95.47
BIGRU%	*Accuracy*	92.37	90.07	92.11	93.52	91.23
	F1 score	94.83	91.76	81.59	87.50	90.77
	detection	92.87	88.02	92.31	95.91	95.82

To verify the effect of the generated samples on the anomaly detection performance, compare to join counter samples before and after the performance difference of anomaly detection model, calculated after injection against sample detection rate and the average accuracy of the four kinds of model, and observation of fig 13 reveals that the adversarial attack causes an overall decrease in the accuracy and detection rate of the four models, and the decline of the seq2seq model and The decrease of the BIGRU model is smaller, and the decrease of the LSTM and GRU models is larger compared with the other two models. The reason for this change is that the added adversarial samples are predicted to be positive cases, that is, FP increases, which leads to the decrease of the accuracy and detection.

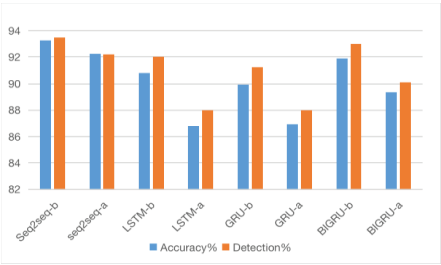


Fig. 13. Comparison of detection performance.

CONCLUSION

This paper proposed an ADS-B malicious sample generation method by using the improved GAN. we construct simulated interference samples and conduct anomaly detection experiments, and generate new attack samples using CTGAN. Through multi-perspective experiments, we demonstrate that CTGAN can generate effective network attack samples and can make the performance of the machine learning model for anomaly detection of ADS-B data reduced.

REFERENCES

[1] Strohmeier, M.; Lenders, V.; Martinovic, I.Security of ADS-B: State of the Art and Beyond. IEEE Communications Surveys & Tutorials. 2013; 17(2).

[2] H, Yang.; Q, Zhou. R, Lu.; H, Li. A Practical and Compatible Cryptographic Solution to ADS-B Security, in IEEE Internet of Things Journal, vol. 6, no. 2, April 2019; pp. 3322-3334.

[3] A, Costin.; A, Francillon.Ghost in the air (traffic): on insecurity of ads-b protocol and practical attacks on ads-b devices Black Hat.USA. 2012, pp.1-12.

[4] Ji, Shouling.; Du, Tianyu.; Li, Jinfeng. A Review of Machine Learning Model Security and Privacy [J]. Journal of Software. 2021,32(01):41-67.DOI:10.13328/j.cnki.jos.006131

[5] E, Habler.; A, Shabtai. Using lstm encoder-decoder algorithm for detecting anomalous ads-b messages Computers & Security,78. 2018, pp.155-173.

[6] Sefi, Akerman.; Edan, Habler.; Asaf, Shabtai. VizADS-B: Analyzing Sequences of ADS-B Images Using Explainable Convolutional LSTM Encoder-Decoder to Detect Cyber Attacks. CoRR. 2019,abs/1906.07921.

[7] Jiang, Y.; G, ZHANG. A review of countermeasures against attack and defense based on deep learning model [J]. Computer engineering,2021,47(01):1-11.DOI:10.19678/j.issn.1000-3428.0059156.

[8] Cai, Xiuxia.; Du, Huimin. Journal of Xi 'an University of Posts and Telecommunications. 2021,26(01):67-75.DOI:10.13682/j.issn.2095-6533.2021.01.011.

[9] Szegedy, C.; Zaremba, W.; Sutskever, I. Intriguing properties of neural networks. arXiv preprint arXiv:1.

[10] Goodfellow, I.; Shlens, J.; Szegedy, C. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572, 2014.

[11] Xiao, C.; Li, B.; Zhu, J-Y. Generating adversarial examples with adversarial networks. arXiv preprint arXiv:1801.02610, 2018.

[12] Yang, Guofeng.; Ding, Yu.; Lu, Naiyan. Insights Into the Two-Dimensional MoS2 Grown on AlGaN(GaN) Substrates by CVD Method. IEEE PHOTONICS JOURNAL. 2021, 13(6).

[13] J, Nilesh.; V, Ramakrishnan. A comparative analysis and an optimized structure of vertical GaN floating gate trench MOSFET for

[14] S, Ansith.; A, Bini. Land use classification of high resolution remote sensing images using an encoder based modified GAN architecture. Displays. 2022,74.

[15] Kojima, Hiroki.; Ikegami, Takashi. Organization of a Latent Space structure in VAE/GAN trained by navigation data. Neural Networks. 2022,152.

[16] Niall.; Adams. Dataset Shift in Machine Learning. Journal of the Royal Statistical Society: Series A (Statistics in Society). 2010,173(1).

[17] Y, Zhang.; N, Zaidi.; J, Zhou. GANBLR: A Tabular Data Generation Model. 2021 IEEE International Conference on Data Mining (ICDM). 2021, 181-190.

[18] Zou, yunkai. Research on ADS-B spoofing interference detection method based on machine learning. Civil Aviation University of China. 2020.

[19] J, Moon.; S, Jung.; S, Park. Conditional Tabular GAN-Based Two-Stage Data Generation Scheme for Short-Term Load Forecasting. IEEE Access. 2020,8,205327-205339.

[20] C, An.; J, Sun.; Y, Wang. A K-means Improved CTGAN Oversampling Method for Data Imbalance Problem. 2021 IEEE 21st International Conference on Software Quality. 2021, 883-887.