Mathematical Biosciences and Engineering

MBE, 19(2): 2030–2042.
DOI: 10.3934/mbe.2022095
Received: 30 September 2021
Accepted: 23 November 2021
Published: 27 November 2021

http://www.aimspress.com/journal/MBE

Research article

Research on hybrid intrusion detection method based on the ADASYN and ID3 algorithms

Yue Li*, Wusheng Xu, Wei Li, Ang Li and Zengjin Liu

School of Computer Science and Technology, Donghua University, Shanghai 201620, China

* **Correspondence:** Email: frankyueli@dhu.edu.cn; Tel: +862167792809.

Abstract: Intrusion detection system plays an important role in network security. Early detection of the potential attacks can prevent the further network intrusion from adversaries. To improve the effectiveness of the intrusion detection rate, this paper proposes a hybrid intrusion detection method that utilizes ADASYN (Adaptive Synthetic) and the decision tree based on ID3 algorithm. At first, the intrusion detection dataset is transformed by coding technology and normalized. Subsequently, the ADASYN algorithm is applied to implement oversampling on the training set, and the ID3 algorithm is employed to build a decision tree model. In addition, the model proposed by the research is evaluated by accuracy, precision, recall, and false alarm rate. Besides, a performance comparison is conducted with other models. Consequently, it is found that the combined model based on ADASYN and ID3 decision tree proposed in this research possesses higher accuracy as well as lower false alarm rate, which is more suitable for intrusion detection tasks.

Keywords: intrusion detection; decision tree; network security; machine learning

1. Introduction

The rapid development of internet of things (IoT) and artificial intelligent (AI) has brought convenience and evolution to the people. Industry 4.0 with the smart factory concept has quickly developed and been deployed since 2011 [1]. Industry 5.0 is currently conceptualized to leverage the creativity of human experts in collaboration with efficient, smart, and accurate machines [2]. Unfortunately, fast development of network intrusion technology has caused insecurity and undermined the reliability of Internet services. As a result, virtual assets and network connected devices have become intrusion targets by attackers and criminals. Therefore, advanced technologies

such as Blockchain and federated learning (FL) are adopted to enhance security service for industrial internet of things (IIoT) and internet of medical things (IoMT, a crucial IoT segment for collect and analyze health data) [3–6].

Intrusion detection system (IDS) is a very important security component to protect the system [7]. Unlike other network security technologies, IDS is an active security protection technology, which can be a software system or a hardware device and can collect information from various systems or network resources and then analyze the characteristics of network traffic to deal with network attacks [8].

IDS can be divided into two main sub-categories as follows: IDS based on misuse and IDS based on anomaly [9]. Both models have different advantages and disadvantages. The former performs a simple pattern matching technique to match an unknown pattern with a known pattern and then considers whether it is normal. The misuse detection can provide high accuracy because it can match predefined attack behaviors in the database. However, if an unknown attack or an attack that does not match any signature is performed, these attacks cannot be detected; the latter applies statistical methods to analyze abnormal behaviors from normal behaviors. Therefore, novel attacks can be detected using this method, but anomaly detection owns a lower accuracy and a higher false alarm rate.

This paper focuses on an anomaly detection model using machine learning algorithms. The experiment adopts the real network traffic data set UNSW-NB15. First, the data preprocessing techniques (such as one-hot encoding, normalization) are applied to process network log data into the data easier to model. Subsequently, the study proposes to detect network attacks based on the combination of a certain oversampling method and Decision Tree. The oversampling method is used to solve the problem of imbalance between the normal data and the abnormal data. Moreover, the Decision Tree algorithm is adopted to classify the traffic into two categories to realize network attack traffic detection. At the same time, to verify the advantages of the model, the proposed method is compared with other machine learning algorithms as well as the combinations in various algorithms.

The rest of the paper is organized as follows: Section 2 presents some previous related work; Section 3 gives a description and analysis of the UNSW-NB15 dataset; Section 4 discusses the proposed methodology; Section 5 presents the experimental results. In the end, the conclusion and future work are given in Section 7.

2. Related works

Many related works in the literature focused on anomaly detection based on various machine learning and data mining techniques.

N. Moustafa et al. [10] applied visual studio business intelligence 2008 to test five machine learning algorithms' performance for the UNSW-NB15 dataset. Namely, they are decision tree, logistic regression, native Bayies, ANN and EM clustering. All of the five algorithms adopt the tool's default parameters, and the decision tree with the optimal performance reveals the highest accuracy rate of 85.56% with the lowest false alarm rate of 15.78%.

V. Kanimozhi et al. [11] applied the recursive feature elimination (RFE) algorithm in the feature selection technology to extract the four most relevant features; utilized artificial neural networks was adopted in the modeling. Their research recursively established a model through applying the optimal attributes. The experiment results show the detection accuracy reaches 89%.

X. P. Tan et al. [12] proposed a method of using the synthetic minority oversampling technique (SMOTE) to balance the dataset and then uses the random forest algorithm to train the classifier for intrusion detection. It achieved an accuracy of 92.57% and improved detection efficiency by reducing computing resources' consumption significantly.

A. P. Muniyandi et al. [13] implemented a semi-supervised learning technique. For this study, they first employed unsupervised learning through k-means clustering, where the percentage of the training instances was trained by using the Euclidean distance method. Subsequently, the supervised learning was performed by applying the C4.5 algorithm. Along with clustering, the boundary was refined, which greatly assisted the C4.5 algorithm in detecting anomalies with much more accuracy.

G. Kim et al. [14] introduced a hybrid detection method that hierarchically integrates the misuse and anomaly detection model. At first, the C4.5 Decision Tree was implemented to train the dataset. After that, they are decomposed into various subsets. Then, SVM was applied to establish the profiles of the normal and abnormal behavior. The experimentation was performed on the NSL-KDD dataset. This hybrid approach displays better performance than the conventional models. However, there is a limitation in this way. Namely, C4.5 will be degraded while decomposing the data into subsets in misuse detection.

S. T. Miller et al. [15] proposed an approach to classify intrusion. This approach is called multi-perspective machine learning (MPML), whose main aim is to improve the accuracy of malware detection through the application of the carefully-selected malware characteristics (represented by different subsets of features). These features are subsequently applied to train classifiers whose results are then combined to give a final prediction. The initial results on the NSL-KDD dataset revealed at least a 4% improvement in contrast to other ensemble methods (such as bagging boosting rotation forest and random forest).

From the current situation of international research, the research goals of intrusion detection systems focus on improving the accuracy of the model, the detection rate, and reducing the false alarm rate. In addition, the data processing methods, such as feature selection methods or data dimensionality reduction methods, are applied to improve model performance and reduce the consumption of computing resources. Finally, the solution to the imbalance of data set sample categories is also an important research direction.

3. Dataset analysis

This research applied the public data set UNSW-NB15 created by the Australian Centre for Cyber Security (ACCS) laboratory in 2015, which contains real normal and abnormal traffic data 16. Forty-nine features are extracted from Pcap files to reflect the nature of network traffic. In contrast to the traditional KDD99 data set and NSLKDD data set, this data set covers normal activities and attack activities within two weeks (including 1 normal type and 9 attack types), reflecting the contemporary network traffic characteristics as well as the new low footprint attack scenarios. In this way, this dataset is more suitable for the current network environment.

In CSV (Comma-Separated Values) files, the total number of records is 2,540,044, which are stored in the four CSV files. Furthermore, a partition from this dataset is configured as the standard training set and the testing set. The number of records in the training set is 175,341 records, and the testing set is 82,332 records [17. Unlike the KDD and NSLKDD datasets, the UNSW-NB15 dataset contains one normal type and nine attack types, no matter whether it is in the training set or in the

test set. The feature distribution of the data set is shown in the Table 1.

The UNSW-NB 15 data set's involved features are classified into six groups as follows: flow features, basic features, content features, time features, additional generated features, and labelled features. Among them, the additional generated features are further categorized into two subgroups called general purpose features and connection features. The following table reflects the distribution of the dataset features [18.

Table 1. Features of UNSW-NB15.

No.	Feature Name	No.	Feature Name
Flow Features		26	res_bdy_len
1	srcip	**Time Features**	
2	sport	27	sjit
3	dstip	28	djit
4	dsport	29	stime
5	proto	30	ltime
Base Features		31	sintpkt
6	state	32	dintpkt
7	dur	33	tcprtt
8	sbytes	34	synack
9	dbytes	35	ackdat
10	sttl	**General purpose features**	
11	dttl	36	is_sm_ips_ports
12	sloss	37	ct_state_ttl
13	dloss	38	ct_flw_http_mthd
14	service	39	is_ftp_login
15	sload	40	ct_ftp_cmd
16	dload	**Connection features**	
17	spkts	41	ct_srv_src
18	dpkts	42	ct_srv_dst
Content Features		43	ct_dst_ltm
19	swin	44	ct_src_ ltm
20	dwin	45	ct_src_dport_ltm
21	stcpb	46	ct_dst_sport_ltm
22	dtcpb	47	ct_dst_src_ltm
23	smeansz	**Labelled Features**	
24	dmeansz	48	attack_cat
25	trans_depth	49	Label

It is important to note that the features scrip(#1), sport(#2), dstip(#3), dsport(#4), stime(#29) and ltime(#30) in Table 1 are dropped in the training and testing data set.

In the labeled features, the 48_{th} dimension data indicates whether the record is normal behavior or an attack, which is divided into one normal type and nine attack types. The 49_{th} dimension data belongs to the binary type, where ***1*** represents attack behavior, and ***0*** represents normal behavior.

According to the above analysis, the following data preprocessing techniques are incorporated into the next experimental tasks.

One-Hot encoding is applied to convert features of nominal types, such as proto(#5), service(#6), state(#14), to numerical features.

The categorical values are transformed in attack_cat(#48) column in training and testing sets to numerical values. In addition, the 48th column should be dropped when performing binary classification tasks.

In order to normalize all numerical features, min-max normalization will be adopted in this study.

After the data preprocessing, the data in the 48th dimension (#48) is based on a nominal expression. In this study, the statistical analysis on each attack type is conducted to explore the sample equilibrium.

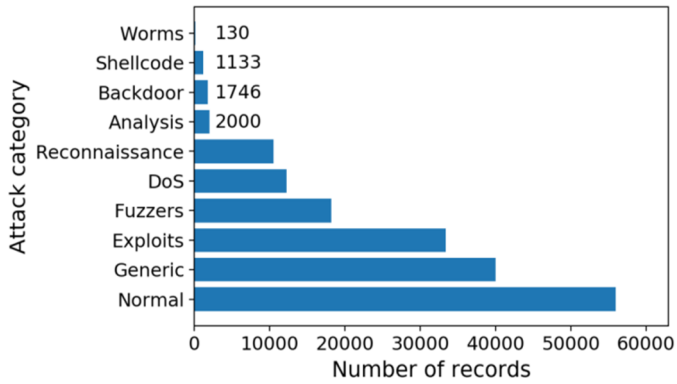

Figure 1. The statistics of sample label.

From the above figure, it is found that there is a serious imbalance in the data label distribution of the 48th dimension. As many as 56,000 data records are representing normal behavior. However, in the abnormal behavior, the number of various attacks is also unevenly distributed. The generic abnormal records have 40,000 records, while the smallest worm records are only 130. The sample imbalance ratio reached 1:430. In this way, it is almost impossible to achieve the high-precision detection of minority classes only through model optimization. Therefore, the 49_{th} dimension data provides the research conditions for binary classifications.

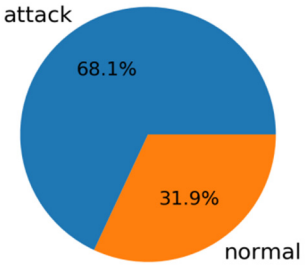

Figure 2. Statistics on the proportion of normal and attack.

Figure 2 reveals the statistics of the 49_{th} dimension. It can be seen that when all the abnormal behaviors are merged into one category, the majority category under the multi-category condition becomes the minority category under the binary-category condition. There is still a sample imbalance between normal behavior and aggressive behavior. Therefore, solving this problem becomes a significant aspect of this research.

4. Proposed intrusion detection method

This section describes the methods applied in this paper for network anomaly detection.

4.1. ADASYN

Among the processing methods of imbalanced data, the common methods are under-sampling and over-sampling. The over-sampling method is applied in this research, and the specific algorithm we adopt is the ADASYN algorithm.

The ADASYN algorithm is an adaptive synthetic sampling approach [19]. Its essential idea is to assign weights to different minority class examples according to their difficulty levels in learning. In contrast to those examples of minorities that are easier to learn, these minority examples will generate more comprehensive data. The sampling rate can achieve a relatively balanced effect, reducing the problem of data imbalance.

We suppose that training dataset D with m samples $\{xi,yi\}$, $I = 1, ..., m$, where xi is an instance in the n-dimensional feature space X and $Y = \{0,1,...\}$ is the class identity label associated with xi. Define ms and ml is defined as the number of minority class examples and the number of majority class examples, respectively. Therefore, $m_s \leq m_l$ and $m_s + m_l = m$.

The steps of the ADASYN algorithm are as follows:

(1) The degree of class imbalance is calculated as follows:

$$d = m_s/m_l\,, d \in (0,1] \tag{1}$$

(2) When $\mathrm{d} < \mathrm{d_{th}}$, the next algorithm steps are continued to follow. First, the number of the synthetic data examples that need to be generated for the minority class is firstly calculated as follows:

$$G = (m_l - m_s) \times \beta, \beta \in [0,1] \tag{2}$$

where β is a parameter applied to specify the desired balance level after the generation of the synthetic data, $\beta = 1$ means that a fully balanced dataset is created after the generalization process.

(3) Secondly, for each sample x_i of the minority class, it is found that their K nearest neighbors based on the Euclidean distance in the n-dimensional space. The ratio r_i defined is calculated as follows:

$$r_i = \Delta_i/K\,, i = 1, \ldots, m_s \tag{3}$$

where, Δ_i is the number of examples in the K nearest neighbors of x_i that belongs to the majority class. Therefore, $r_i \in [0,1]$.

(4) Next, r_i from Eq (3) is normalized as follows:

$$\hat{r}_i = r_i / \sum_{i=1}^{m_s} r_i \quad (4)$$

So that $\hat{r}_i$ is a density distribution, implying that $\sum_i \hat{r}_i = 1$.

(5) Then, the number of synthetic data examples that need to be generated for each minority example is calculated as follows:

$$g_i = \hat{r}_i \times G \quad (5)$$

where G is the total number of synthetic data examples, it needs to be generated for the minority class as defined in Eq (2).

(6) Finally, g_i synthetic samples for each minority class data example x_i is generated according to the following steps:

Do the *Loop* from 1 to g_i :

A minority sample x_{zi} from the K nearest neighbors of x_i is randomly selected;

The synthetic data example s_j is generated according to the following formula, where λ is a random number, λ∈ [0,1].

$$s_j = x_i + (x_{zi} - x_i) \times \lambda \quad (6)$$

EndLoop

There are 175,341 training samples in the data set applied in this research, including 56,000 normal samples and 119341 attack samples. Therefore, the research needs to oversample the normal category and set β to 1 to create a fully balanced dataset, while the ADASYN algorithm needs to generate 63,341 new records. The important role played by ADASYN in our proposed method is shown in Figure 3.

4.2. Decision tree

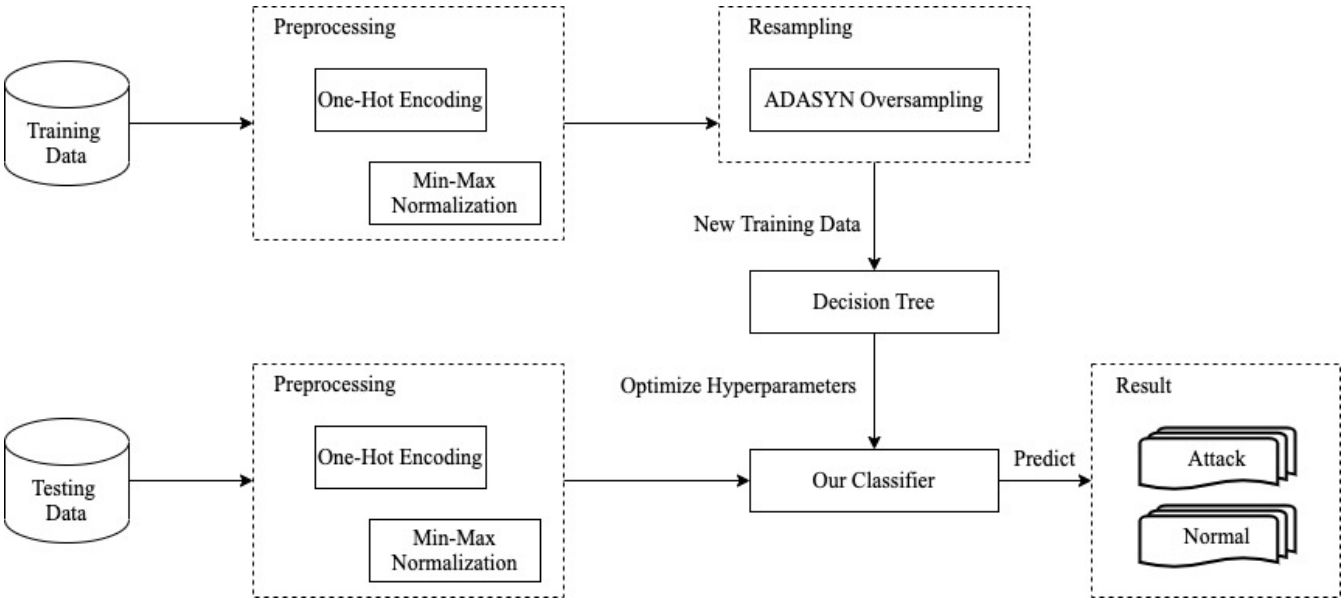

Figure 3. The procedures of proposed intrusion detection method.

Decision Tree is one of the classification algorithms in data mining that makes use of a tree-like structure to perform a decision 21. Usually, Decision Tree are applied in operation research and intrusion detection. The well-known methods for automatically building Decision Tree are the ID3 21 and C4.5 23 algorithms. Both algorithms establish Decision Tree from a set of training data through applying the concept of information entropy. The difference between them is that ID3 mainly adopts information gain for feature selection, while C4.5 adopts information gain ratio. In this research, the ID3 algorithm is adopted.

The ID3 algorithm will select the optimal feature (the feature with the largest information gain) for node generation 24. The process of ID3 is specified as follows.

(1) S is supposed to be a dataset. The class label attribute assume to have n different values, and the definition of n have different classes $C_i(i = 1, \ldots, n)$. S_i is set to be the number of samples in class C_i. Then, the entropy is calculated as follows:

$$Entropy(\mathrm{S}) = -\sum_{i=1}^{n} p_i log_2(p_i) \tag{7}$$

where, p_i is the probability of any sample belonging to C_i, $p_i = S_i/S$.

(2) The attribute A is supposed to have k different values in the dataset S. According to attribute A, S is divided into k sample subsets $\{S_1, \ldots, S_k\}$. Then, the information entropy of the sample subset after dividing S by attribute A is calculated as follows:

$$Entropy_A(S) = \sum_{i=1}^{k} \frac{|S_i|}{|S|} Entropy(S_i) \tag{8}$$

where, $|S_i|$ is the number of samples included in the sample subset S_i, and $|S|$ is the number of samples included in the sample set S.

(3) It is assumed that the dataset S is divided according to attribute A, then the calculation method of the information gain is as follows:

$$Gain(S, A) = Entropy(\mathrm{S}) - Entropy_A(S) \tag{9}$$

Calculate the information gain of each attribute in the data set S is calculated in turn. The larger the information gain of a certain attribute is, the purer the sample subset divided by the attribute is, and the better it is for classification. The attribute with the greatest information gain in each step will be applied.

As shown in Figure 3, this study not only applied the ID3 algorithm but also applied the pruning operation and hyperparameter optimization in order to achieve the best performance, limiting the maximum depth of the decision tree, the minimum number of samples for split, and the minimum number of samples at a leaf node.

5. Results and discussions

5.1. *Evaluation measures*

The confusion matrix and various evaluation measures will be described in this section. The main parameters of the confusion matrix are as follows 25.

TN (true negative): The real is normal traffic, and the model classification result is also normal traffic;

FN (false negative): The real is attack traffic, while the model classification result is normal traffic;

TP (true positive): The real is attack traffic, and the model classification result is also attack traffic;

FP (false positive): The real is normal traffic, while the model classification result is attack traffic;

Table 2. Confusion matrix.

		Predicted	
		Attack	**Normal**
Actual	**Attack**	TP	FN
	Normal	FP	TN

The statistical indicators, such as accuracy, precision, recall, and false alarm rate (FAR) will be applied in this study to evaluate and compare the performance of the model. The calculation methods are as follows.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{10}$$

$$Precision = \frac{TP}{TP+FP} \tag{11}$$

$$Recall = \frac{TP}{TP+FN} \tag{12}$$

$$FAR = \frac{FP}{TN+FP} \tag{13}$$

5.2. *Performance evaluation*

The confusion matrix from the combination of ADASYN and the ID3 on the test set is as follows. Subsequently, the study calculated various measurement indicators through the results.

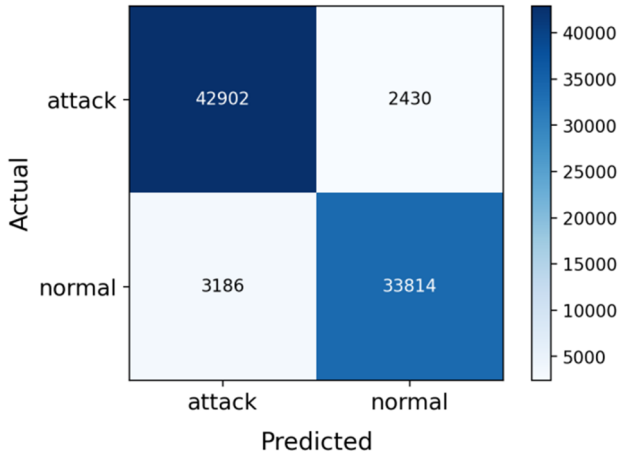

Figure 4. Result of confusion matrix.

In order to evaluate the performance of various classification algorithms on UNSW_NB15 dataset for the binary classification, K-nearest neighbor (KNN), logistic regression, support vector machine classifier (SVC), random forest, adaboost, decision tree (based on ID3 algorithm) and our proposed method (ADASYN+ID3) are applied to train models through the training set. Subsequently, the models are applied to the testing set. The results are mentioned in Table 3.

Table 3. Performance comparison of different algorithms.

No.	Approach	Accuracy(%)	Precision(%)	Recall(%)	FAR(%)
1	KNN	85.17	80.36	96.17	28.96
2	Logistic Regression	81.35	76.49	95.48	35.96
3	SVC	87.05	82.95	96.27	24.25
4	RandomForest	90.23	87.17	96.68	17.74
5	Adaboost	89.11	85.51	96.59	20.09
6	DecisionTree (ID3)	88.95	84.62	97.66	21.43
7	ADASYN + ID3	93.18	93.09	94.64	8.61

According to Table 3, it is worth noting that we used the Random Forest algorithm based on the bagging strategy and the Adaboost algorithm based on the boosting strategy, both of which are part of integrated learning, this is to better compare the performance of different types of machine learning algorithms in intrusion detection tasks.

On the other hand, we also introduced the ROC (receiver operating characteristic) curve and AUC (area under the curve) in the result analysis. We will combine the data in Table 3 and the ROC curve and AUC to comprehensively compare the model performance. The ROC curve and AUC are shown in Figure 5.

From the Table 3, it is found that the proposed model performs best in terms of accuracy, precision, and false alarm rate. And in Figure 4, We can found that compared with KNN (No.1), logistic regression (No.2), and SVC (No.3), the proposed model has a more robust ROC curve, which means that our model is more stable. On the other hand, comparing with randomForest (No.4), Adaboost (No.5), decisiontree (No.6), the detection rate of the proposed model is rising fastest in the case of ensuring a very low false alarm rate (less than 2%). These results and analysis mean that our model presented desired performance.

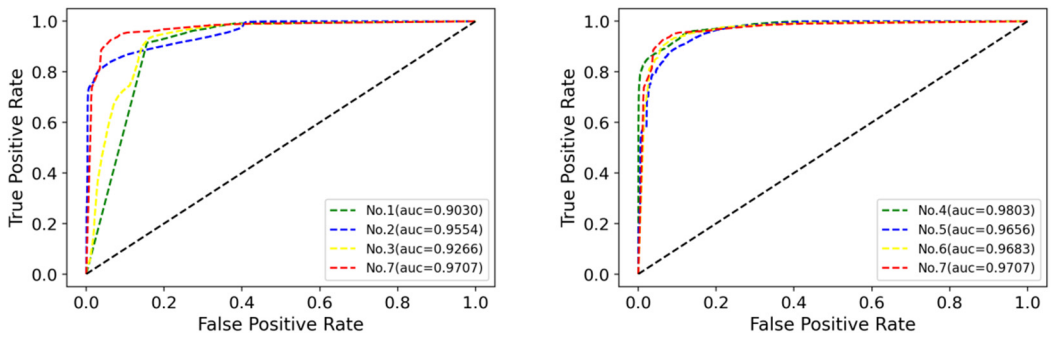

Figure 5. Comparison of ROC and AUC.

In addition, it is worth mentioning that the study also compared the proposed model with the models in other researches. Unfortunately, some of the studies we referred to did not give detailed model performance data, but we still compared the results obtained, where missing data was marked with '–'. The comparison results are shown in Table 4.

Table 4. Performance comparison with other researchers' models.

Approach	Accuracy (%)	Precision (%)	Recall (%)	FAR (%)	Reference
EM Clustering	78.47	–	–	23.79	M. Salem et al. 26
TSDL	89.71	86.70	92.46	12.76	F. A. Khan et al. 26
DEA	92.40	–	–	8.20	N. Moustafa et al. 28
NSNAD+HLA	91.91	94.29	93.80	12.10	N. B. Aissa et al. 29
ADASYN+ID3	93.18	93.09	94.64	8.61	Our Proposed

By comparing with other studies, it is found that the method proposed for intrusion detection tasks achieved the highest accuracy of 93.18% and the hightest recall of 94.64%. Meanwhile, its False Alarm Rate is second only to the DEA method, but their FAR are very close. Therefore, the performance of our proposed model in intrusion detection tasks is very competitive.

6. Conclusions

In this study, machine learning technology is applied to develop an intelligent and efficient intrusion detection system. Furthermore, the calculation process of the decision tree based on the ID3 algorithm is described, and the powerful classification ability of the ID3 algorithm is demonstrated, verifying that the ADASYN oversampling method has a certain effect on the treatment of sample imbalance. In comparison with other intrusion detection classification methods, the combination of ADASYN and ID3 algorithm proposed in this paper possesses a higher accuracy rate with a lower false alarm rate under the task of binary classification on the UNSW-NB15 dataset.

This study's future research work will focus on model fusion, which can provide higher accuracy for intrusion detection tasks. On the other hand, while applying model fusion technology, the feature selection or feature dimensionality reduction technology suitable for intrusion detection tasks will be studied to enhance intrusion detection systems' performance and reduce the time required to detect attacks.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under grand No. 61772129.

Conflict of interest

The authors declared that they have no conflicts of interest to this work.

References

1. Y. Lu, Industry 4.0: A survey on technologies, applications and open research issues, *J. Ind. Inf. Integr.*, **6** (2017), 1–10. doi: 10.1016/j.jii.2017.04.005.
2. P. K. Maddikunta, Q. Pham, P. Deepa, K. Dev, T. R. Gadekallu, R. Ruby, et al., Industry 5.0: A survey on enabling technologies and potential applications, *J. Ind. Inf. Integr.*, **2021** (2021). doi: 10.1016/j.jii.2021.100257.
3. W. Wang, H. Xu, R. Gadekallu, Z. Han, C. Su, Blockchain-based reliable and efficient certificateless signature for IIoT devices, *IEEE Trans. Ind. Inf.*, **2021** (2021), 1551–3203. doi: 10.1109/TII.2021.3084753.
4. H. Xiong, C. Jin, M. Alazab, K. H. Yeh, H. Wang, T. R. R. Gadekallu, et al., On the design of Blockchain-based ECDSA with fault-tolerant batch verication protocol for Blockchain-enabled IoMT, *IEEE J. Biomed. Health Inf.*, **2021** (2021). doi: 10.1109/JBHI.2021.3112693.
5. W. Wang, C. Qiu, Z. Yin, G. Srivastava, T. R. Gadekallu, F. Alsolami, et al., Blockchain and PUF-based lightweight authentication protocol for wireless medical sensor networks, IEEE *Internet Things J.*, **2021** (2021). doi: 10.1109/JIOT.2021.3117762.
6. W. Wang, M. H. Memon, Z. Lian, Z. Yin, Q. V. Pham, T. R. Gadekallu, et al., Secure-enhanced federated learning for ai-empowered electric vehicle energy prediction, *IEEE Consum. Electron. Mag.*, **2021** (2021). doi: 10.1109/MCE.2021.3116917.
7. W. Lee, S. J. Stolfo, K. W. Mok, A data mining framework for building intrusion detection models, in *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, 1999. doi: 10.1109/SECPRI.1999.766909.
8. A. Buczak, E. Guven, A survey of data mining and machine learning methods for cyber security intrusion detection, *IEEE Commun. Surv. Tutorials*, **18** (2016), 1153–1176. doi: 10.1109/COMST.2015.2494502.
9. P. Animesh, J. Park, An overview of anomaly detection techniques: Existing solutions and latest technological trends, *Comput. Networks*, **51** (2007), 3448–3470. doi: 10.1016/j.comnet.2007.02.001.
10. N. Moustafa, J. Slay, The evaluation of network anomaly detection systems: statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set, *Inf. Secur. J.*, **25** (2016), 18–31. doi: 10.1080/19393555.2015.1125974.
11. V. Kanimozhi, P. Jacob, UNSW-NB15 dataset feature selection and network intrusion detection using deep learning, *Int. J. Recent Technol. Eng.*, **7** (2019), 443–446. doi: 10.1080/19393555.2015.1125974.
12. X. P. Tan, S. J. Su, Z. P. Huang, X. J. Guo, Z. Zuo, X. Sun, et al. Wireless sensor networks intrusion detection based on SMOTE and the random forest algorithm, *Sensors*, **19** (2020), 203. doi: 10.3390/s19010203.
13. A. Muniyandi, R. Rajeswari, R. Rajaram, Network anomaly detection by cascading k-Means clustering and C4.5 decision tree algorithm, *Proc. Eng.*, **30** (2012), 174–182. doi: 10.1016/j.proeng.2012.01.849.
14. G. Kim, S. Lee, S. Kim, A novel hybrid intrusion detection method integrating anomaly detection with misuse detection, *Expert Syst. Appl.*, **41** (2014), 1690–1700. doi: 10.1016/j.eswa.2013.08.066.

15. S. Miller, C. Busby-Earle, Multi-perspective machine learning a classifier ensemble method for intrusion detection, in *Proceedings of the 2017 international conference on machine learning and soft computing*, **2017** (2017), 7–12. doi: 10.1145/3036290.3036303.
16. N. Moustafa, J. Slay, UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set), in *2015 Military Communications and Information Systems Conference, MilCIS 2015-Proceedings*, 2015. doi: 10.1109/MilCIS.2015.7348942.
17. Australian Centre for Cyber Security (ACCS), The UNSW-NB15 Dataset Description. Available from: https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/.
18. T. Janarthanan, S. Zargari, Feature selection in UNSW-NB15 and KDDCUP'99 datasets, in *2017 IEEE 26th International Symposium on Industrial Electronics (ISIE)*, (2017), 1881–1886. doi: 10.1109/ISIE.2017.8001537.
19. H. He, Y. Bai, E. A. Garcia, S. Li, ADASYN: Adaptive synthetic sampling approach for imbalanced learning, in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, (2008), 1322–1328. doi: 10.1109/IJCNN.2008.4633969.
20. P. Liu, M. Hong, D. Huang, Y. Luo, S. Wang, Joint ADASYN and AdaBoostSVM for imbalanced learining, *J. Beijing Univ. Technol.*, **43** (2017), 368–375.
21. J. R. Quinlan, Induction of Decision Tree, *Machine Learning*, **1** (1986), 81–106. doi: 10.1007/BF00116251.
22. X. Wang, L. Wang, N. Li, An application of decision tree based on ID3, *Phys. Procedia*, **25** (2012), 1017–1021. doi: 10.1016/j.phpro.2012.03.193.
23. J. R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers Inc., 1992.
24. J. R. Quinlan, Decision tree and decision-making, *IEEE Trans. Syste. Man Cybern.*, **20** (1990), 339–346. doi: 10.1109/21.52545.
25. R. Susmaga, Confusion matrix visualizatio, in *Intelligent Information Processing and Web Mining*, Springer, Berlin, Heidelberg, (2004), 107–116. doi: 10.1007/978-3-540-39985-8_12.
26. M. Salem, U. Buehler, Mining techniques in network security to enhance intrusion detection systems, *Int. J. Network Secur. Its Appl.*, **2012** (2012), 167–172. doi: 10.5121/ijnsa.
27. F. A. Khan, A. Gumaei, A. Derhab, A. Hussain, A novel two-stage deep learning model for efficient network intrusion detection, *IEEE Access*, **7** (2019), 30373–30385. doi: 10.1109/ACCESS.2019.2899721.
28. A. L. H. Muna, N. Moustafa, E. Sitnikova, Identification of malicious activities in industrial Internet of things based on deep learning models, *J. Inf. Secur. Appl.*, **41** (2018), 1–11. doi: 10.1016/j.jisa.2018.05.002.
29. M. Guerroumi, A. Derhab, NSNAD: negative selection-based network anomaly detection approach with relevant feature subset, *Neural Comput. Appl.*, **32** (2020), 3475–3501. doi: 0.1007/s00521-019-04396-2.

AIMS Press