A Self-supervised Adversarial Learning Approach for Network Intrusion Detection System

Lirui Deng[1], Youjian Zhao[1,2](✉), and Heng Bao[3]

[1] Department of Computer Science and Technology, Tsinghua University,
Beijing, China
dlr18@mails.tsinghua.edu.cn, zhaoyoujian@tsinghua.edu.cn
[2] Zhongguancun Laboratory,Beijing, China
[3] School of Cyber Security, University of Chinese Academy of Sciences,
Beijing, China
baoheng@iie.ac.cn

Abstract. The network intrusion detection system (NIDS) plays an essential role in network security. Although many data-driven approaches from the field of machine learning have been proposed to increase the efficacy of NIDSs, it still suffers from extreme data imbalance and the performance of existing algorithms depends highly on training datasets. To counterpart the class-imbalanced problem in network intrusion detection, it is necessary for models to capture more representative clues within same categories instead of learning from only classification loss. In this paper, we proposed a self-supervised adversarial learning approach for intrusion detection, which utilize instance-level discrimination for better representation learning and employs a adversarial perturbation styled data augmentation to improve the robustness of NIDS on rarely seen attacking types. State-of-the-art result was achieved on multiple frequently-used datasets and experiment conducted on cross-dataset setting demonstrated good generalization ability.

Keywords: Network intrusion detection · Self-supervise learning · Adversarial learning

1 Introduction

While the advent of the Internet has brought immense convenience to our daily lives in recent decades, it has also unavoidably introduced dozens of new challenges. As people nowadays spend more time in cyberspace than real world no matter living or working, attacking on network activities with various kinds of intrusion techniques to prey privacy information or corporation confidential information has never stop. Therefore, as a counterpart, the intrusion detection system (IDS) which safeguard the integrity and availability of key assets has always

W. Lu et al. (Eds.): CNCERT 2022, CCIS 1699, pp. 73–85, 2022.
https://doi.org/10.1007/978-981-19-8285-9_5

been a hot research topic in computer and network security community. In contrast to host-based IDS which are distributed at end point users' system, network intrusion detection system (NIDS) primarily characterized as a solution inside the data transfer pipeline between computers that can monitor the network traffic and alert or even take active response measures when malicious behavior is spotted [4]. Other than some NIDS designed for specific network environment [16,19,24] like Hadoop-based platforms or particular cloud system, most general NIDS researches [10,14,33,38] were performed on network intrusion detection datasets to demonstrate and compare their effectiveness and generalization ability in a data-driven fashion.

Among several limitations of existing algorithms, data imbalance in different classes, especially the lack of data in rarely seen attacking categories, is a one of the most challenging problems. However, it is also a very common phenomenon in network intrusion detection datasets considering the difficulty in data collection or generation. Benign traffic is no doubt the majority part of internet data transfer, not to mention the inherent nature of malicious network activity as of being disguised. While the performance of most traditional ML-based method declines significantly in the case of learning from imbalanced data, a large amount of researches try to address this problem by various approaches [5,8,20,28,34,36,38]. Recently, contrastive learning has drawn a lot of attention with impressive performance improvement [27,35] in computer vision and natural language processing. Besides supervised contrastive learning, instance-level discrimination framework in self-supervised fashion have also shown promising result with few-Shot classification [21] and quickly being used in NIDS research [22].

Inspired by the success of contrastive learning and adversarial learning in CV and NLP, in this paper we proposed a self-supervised adversarial learning (SSAL) approach for network intrusion detection. The main contributions of this paper are as follows:

- First, we utilized an adversarial learning approach for NIDS with design of netflow-based adversarial examples, which improves robustness on class-imbalanced datasets by explicitly suppressing the vulnerability in the representation space and maximizing the similarity between clean examples and their adversarial perturbations.
- We proposed a 2-stage pre-train style self-supervised learning in SSAL that leverages instance-level self-supervised contrastive learning and adversarial data augmentation to achieve a better representation over limited sample, which has not been proposed for NIDS to the best of our knowledge.
- We conducted a experimental evaluation with existing methods on multiple datasets including UNSW-NB15 [26] and CIC-IDS-2017/2018 [31], which shows boosted performance of several machine learning baselines across different datasets.

2 Related Work

In this section we summarize the algorithms and research work related to this study.

2.1 Network Intrusion Detection System

Data-driven methods have been developed and deployed for NIDSs for more than two decades [9]. In order to achieve an effective NIDS, various methods including both machine learning (ML) and deep learning (DL) techniques have been proposed by research community.

Traditional machine learning algorithm such as KNN, PCA, SVM, and tree-based models have all been adopted with intrusion detection, and often used as baseline for particular improved module. For example, Gao **et al.** [11] used classification and regression trees (CARTs) on NSL-KDD datasets with a ensemble scheme where multiple trees were trained on adjusted sampling. Karatas **et al.** [17] addressed the dataset imbalance problem by reducing the imbalance ratio using Synthetic Minority Oversampling Technique (SMOTE), and used different ML algorithms as a baseline for cross comparison that shows improved detection ability for minority class attacks.

Recent studies suggested that the use of DL algorithms for NIDSs have much superior performance than the ML-based methods. RNN and autoencoder [1] was pointed to be the most frequently used models for NIDS in past decades. Regarding data imbalance, Yu **et al.** proposed a CNN-based few shot learning model to improve the detection reliability of network attack categories with the few sample problem. Manocchio proposed FlowGAN [23] which utilized generative models for data augmentation. However, most DL schemes are more complex and require extensive computing resources compare to ML-based methods.

2.2 NIDS Datasets

High-quality data sets are definitely required to fully evaluate the performance of various intrusion detection systems. Many contributions have been published in recent years containing representative network flow data with different kinds of preproccess, which are provided mainly in three categories of formats.

Packet Based Data. The most original and commonly used format is packet based data captured in pcap format and contains payload. Early NIDS datasets does not provide packet based data because it takes too much storage space. But datasets published more recently like CIC-IDS-2017/2018, UNSW-NB15 and LITNET-2020 [7] tend to provide both pcap files and flow based features for the benefit of comparison between different NIDS methods.

Flow Based Data. Flow based data is much more condensed compare to packet based data. It aims to describes the behavior of whole network connection session by aggregate all packets sharing same properties within a time window. Commonly used flow-based formats includes NetFlow [6], OpenFlow [25] and NFStream [2]. CICFlowmeter (formerly known as ISCXFlowMeter [32]) is another important network flow format generator, which tranfers pcap files into more than 80 netflow features, since it was published by Canadian Institute for Cybersecurity therefore used by both CICIDS-2017 and CICIDS-2018.

Other Data. This summarize all data sets that are neither purely packet-based nor flow-based. For example, The KDD CUP 1999 [18] contains host-based attributes like number of failed logins, which can only obtained from above network interface. As a consequence, dataset of this category has its own set of attributes and can not be unified with each other.

2.3 Contrastive Learning

Contrastive learning techniques has been widely used in metric learning such as triplet loss [30] and contrastive loss [13]. While in recent self-supervised approaches, contrastive learning mostly shares a core idea of minimizing various kinds of contrastive loss (i.e. NCE [12], infoNCE [27]) evaluated on pairs of data augmentations. Typically, augmentations are obtained by data transformation (i.e. rotation, cropping, color Jittering in CV, or masking in NLP), but using "adversarial augmentations" as challenging training pairs that maximize the contrastive loss shows more robustness in recently study [15].

3 Approach

In this section, we will explain the main algorithms of our proposed self-supervised adversarial learning framework for data imbalance network intrusion detection.

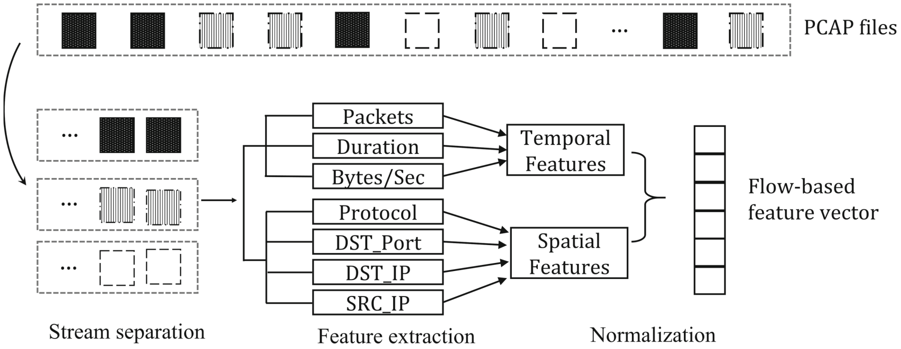


Fig. 1. preprocess pipeline from PCAP files to flow-based feature vector

3.1 Data Preprocessing

To build a comparable cross-dataset evaluation process, we adopt commonly used datasets UNSW-NB15, CIC-IDS-2017 and CIC-IDS-2018, as they not only contain a wide range of attack scenarios but also provide original pcap files that can be easily processed into unified feature set. CIC-IDS-2017 dataset is made up of 5 days network traffic with 7 different network attacking, which forms 51GB size of data. The benign traffic was generated with profile system to protect user privacy. It provides both network traffic (pcap files) and event logs for attack label on each machine. CIC-IDS-2018 dataset is also created by CICFlowMeter but with both benign and malicous profile system, and has more than 400GB pcap data among 17 days. UNSW-NB15 was release in 2015 by Australian Centre for Cyber Security (ACCS) that contains a total of 100 GB of pcap files, consist of 2,218,761 (87.35%) benign flows and 321,283 (12.65%) attack ones.

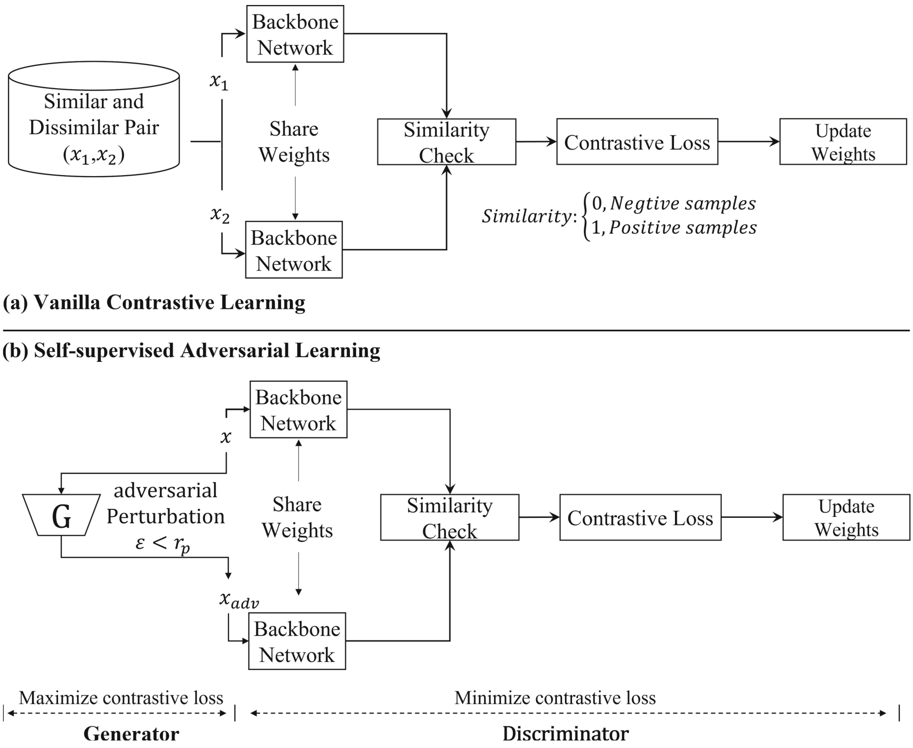
After obtaining original PCAP files, we follow the setting from [29] and take 43 extended feature dimension from the latest *netflow version 9 flow-record format* [6] for flow-based feature extraction (full feature set can be obtained from [29]). Netflow was proposed by Cisco and has become one of the most commonly used flow-based formats for recording network traffic. A network flow stream is an aggregation of a sequence of packets in a continuous session (of TCP connection by default) with the same source IP, source port, destination IP, destination port, and transport protocol. The distribution of our processed unified dataset is shown at Table 1.

Table 1. Distribution of Unified Dataset

	NF-UNSW-NB15	CIC-IDS-2018	CIC-IDS-2017	Summary	Ratio
Benign	2295222	16635567	2359087	21289876	88.29%
Fuzzers	22310	0	0	22310	0.09%
Analysis	2299	0	0	2299	0.01%
Backdoor	2169	0	0	2169	0.01%
DoS	5794	483999	252660	742453	3.08%
Exploits	31551	0	0	31551	0.13%
Generic	16560	0	0	16560	0.07%
Reconnaissance	12779	0	0	12779	0.05%
Shellcode	1427	0	0	1427	0.01%
Worms	164	0	0	164	0.00%
BruteForce	0	120912	15994	136906	0.57%
Bot	0	143097	1966	145063	0.60%
DDoS	0	1390270	41835	1432105	5.94%
Infiltration	0	116361	0	116361	0.48%
Web Attack	0	3502	0	3502	0.01%
portscan	0	0	158930	158930	0.66%

Session stream separation might be a little tricky since streams obtained by only quintuple may not be accurate and contain too much data packets. Inspired by [37], other than following tcp handshake flags, we further segment streams by a timeout mechanism to cut idle stream into more pieces with periodic reset. The procedure of generating NIDS datasets with unified feature set is show in Fig. 1.

3.2 Self-supervised Adversarial Learning


(a) Vanilla Contrastive Learning

(b) Self-supervised Adversarial Learning

Fig. 2. Self-supervised Adversarial Learning vs. Vanilla Contrastive Learning

In self-supervise styled contrastive learning (CL), the dataset $\boldsymbol{D} = \{\mathbf{x}_i\}_{n=1}^{N}$ is unlabeled, and each example $\mathbf{x}_i$ from a mini-batch is either paired with a positive sample $\mathbf{x}_i^{'}$ by transformations $\boldsymbol{T}$ or a negative sample $x_j/x_{j,j\neq i}^{'}$. CL seeks to learn an invariant representation of $\mathbf{x}_i$ by minimizing the distance between positive samples defined as:

$$\mathcal{L}_{\mathrm{CL}} = -\log \frac{\exp(\mathrm{sim}(\mathbf{x}_i, \mathbf{x}_j))}{\sum \exp(\mathrm{sim}(\mathbf{x}_i, \mathbf{x}_k))} \tag{1}$$

While Chen **et al.** demonstrate in SimCLR [3] that a temperature parameter τ and a non-linear projector $\boldsymbol{G}$ after backbone network is crucial to the performance of self-supervise CL, we adopt SimCLR loss $\mathcal{L}_{\mathrm{SimCLR}}$ for the base setting of SSAL:

$$\begin{aligned} \mathcal{L}_{\mathrm{SimCLR}}(\mathbf{x}_i, \mathbf{x}_j) &= -\log \frac{\exp(\mathrm{sim}(\mathbf{z}_i, \mathbf{z}_j)/\tau)}{\sum_{k=1}^{2N} \exp(\mathrm{sim}(\mathbf{z}_i, \mathbf{z}_k)/\tau)}, \\ \text{where} \quad \mathbf{h}_i = f(\mathbf{x}_i), &\quad \mathbf{h}_j = f(\mathbf{x}_j), \\ \text{and} \quad \mathbf{z}_i = g(\mathbf{h}_i), &\quad \mathbf{z}_j = g(\mathbf{h}_j) \end{aligned} \tag{2}$$

Adversarial Attack. The design of positive and negative sampling strategy is key to performance of CL models, and the robustness of model will largely depend on the difficulty of proposed sample pairs. As opposed to vanilla contrastive learning, self-supervise adversarial learning leverages adversarial augmentation to ease the difficulty in hard sample mining. Define the perturbation ϵ using L_∞-Norm attack for example:

$$\epsilon = \arg\max_{||\epsilon||_\infty} \mathcal{L}_{\text{SimCLR}}(\mathbf{x}_i, \mathbf{x}_i + \epsilon) \tag{3}$$

With perturbations ϵ given in certain radius that lead to the most diverse positive pairs, we have a adversarial training scheme by both encouraging the learning algorithm to produce a more invariant representation upon updating parameter θ and then find the ϵ' under θ' again. This pipeline is described in Fig. 2 (Fig. 3).

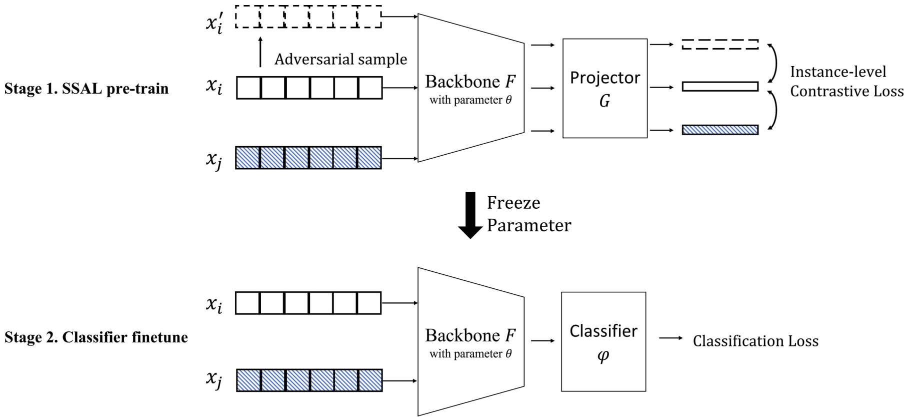


Fig. 3. Framework of proposed 2-stage SSAL NIDS training process

3.3 Classifier Fine-Tune

With SSAL we can already pre-train the model without any class labels in adversarial fashion, but without class annotation pre-trained model cannot be directly used for class-level classification.

Therefore we froze the parameter θ from pre-trained model f, and switch projector head g with a non-linear classifier ψ. The training was conducted under standard multi-class single-label training:

$$\begin{gathered} \mathbf{z}_i = \psi(f(\mathbf{x}_i)), \quad for\ i = 1, 2, \ldots, N \\ p_{i,c} = \sigma(z_{i,c}) = \frac{e^{z_{i,c}}}{\sum_{j=1}^{M} e^{z_{i,j}}}, \quad for\ c = 1, 2, \ldots, M \end{gathered} \tag{4}$$

with cross entropy loss:

$$\mathcal{L}_{ce}(\mathbf{x}_i, \mathbf{l}_i) = -\sum_{c=1}^{M} y_{i,c} \log(p_{i,c}) \tag{5}$$

The full process of proposed 2-stage SSAL for NIDS is shown in Algorithm 1.

Algorithm 1: self-supervised adversarial learning for NIDS

```
**Stage1** *SSAL pre-train*
    input  : Dataset $\boldsymbol{D} = \{\mathbf{x}_i\}_{n=1}^{N}$
    output: model $f$
    Initial model $f$ with parameter $\theta$ and projector $g$
    repeat
        for all $x \in$ minibatch B do
            generate $\epsilon = \arg\max_{||\epsilon||_\infty} \mathcal{L}_{\text{SimCLR}}(\mathbf{x}_i, \mathbf{x}_i + \epsilon)$
            $\theta' = \theta + \nabla_x \mathcal{L}_{\text{SimCLR}}(\mathbf{x}, \mathbf{x} + \epsilon)$
        end
    until reach epoch $N$ or $L \leq \delta_1$
Stage2 Classifier Fine-tune
    input  : Dataset with label $\boldsymbol{D} = \{\mathbf{x}_i, \mathbf{l}_i\}_{n=1}^{N}$, model $f$ with parameter $\theta$
    output: model $f$ and classifier $\psi$
    Initial classifier $\psi$ with parameter $\rho$, freeze $\theta$
    repeat
        for all $x \in$ minibatch B do
            $\rho' = \rho + \nabla_x \mathcal{L}_{ce}(\mathbf{x}_i, \mathbf{l}_i)$
        end
    until reach epoch $N$ or $L \leq \delta_2$
```

4 Experiment Results

Metric and Implementation. The evaluation is conducted by comparing the classifier performance with various classification metrics. The intrusion detection datasets we evaluate on contain several attacking categories, which can be treated as both binary classification and multiple classification problem. While comparing performance under binary classification scenario, the basic terms used in the evaluation is as follow:

$$\begin{aligned} Accuracy(ACC) &= \frac{TP + TN}{TP + FP + TN + FN}, \\ DetectionRate(DR) &= \frac{TP}{TP + FN}, \quad a.k.a \;\; Recall, \\ Precision &= \frac{TP}{TP + FP}, \\ F1Score &= \frac{2 \times Precision \times Recall}{Precision + Recall}. \end{aligned} \tag{6}$$

where TP stands for numbers of true positive samples, FN for false negative, and so forth.

For multi-class classification setting with more detailed label of attacking types, weighted average measure of above metric was adopted considering the proportion for each label in the dataset. To achieve a fair evaluation, five cross-validation splits are conducted and the mean is measured.

Evaluation on Unified Feature Dataset. With the unified feature set upon pre-processed UNSW-NB15 and CIC-IDS-2017/2018 dataset mentioned in Sect. 3.1, we conduct a evaluation across multiple datasets. For the purpose of comparison, we implemented a simple MLP and the Extra Trees model from [29] as baseline models. In Table 2, we can see that our SSAL method achieved outstanding result in all three datasets and exceed previous works in most metrics.

Table 2. Performance on unified dataset

Dataset	Metric	MLP	Extra trees [29]	SSAL
NF-UNSW-NB15	ACC	91.02	**99.73**	99.71
	DR	79.45	97.07	**97.45**
CIC-IDS-2017	ACC	88.42	97.46	**99.57**
	DR	82.61	96.54	**97.14**
CIC-IDS-2018	ACC	83.13	99.35	**99.89**
	DR	76.63	97.12	**98.63**
Overall	ACC	88.1	97.91	**99.63**
	DR	81.6	96.65	**97.35**

Table 3 presents the detailed detection results of different attacking class on the merged NIDS dataset. While using the same backbone (Multi-Layer Perceptron), the performance of model with SSAL pre-train was largely improved on rare seen attacking data.

Table 3. Detailed performance of different classes on unified dataset.(ACC)

ClassName	MLP	SSAL	Class Name	MLP	SSAL
Benign	81.53	99.14	Shellcode	21.73	89.13
Fuzzers	64.31	84.65	Worms	34.86	60.91
Analysis	46.82	82.43	BruteForce	65.69	88.54
Backdoor	49.73	85.77	Bot	61.34	81.17
DoS	54.34	97.63	DDoS	53.52	99.76
Exploits	59.11	93.22	Infiltration	54.43	73.82
Generic	58.27	88.61	Web Attack	62.77	71.44
Reconnaissance	32.92	91.28	portscan	46.98	85.38
Overall	76.63	98.63			

Further Ablation. To further demonstrate the superiority of our proposed method, we compare our method with different backbone networks with ablation studies upon SSAL modules. We first use two different frequently used backbones, MLP and CNN, and plug them with SSAL pre-train for representation learning. The evaluation result shown on Table 4 proves that SSAL can effectively enhance the ability of network intrusion detection systems. As for feature extraction, Table 5 shows the result of different classifiers when SSAL was used as a feature extractor. We first pre-train with all unlabeled training data with SSAL for feature extraction, then freese the network parameter and use SVM or k-NN as a classifier to check the representative ability of SSAL model.

Table 4. Performance with different backbone.(ACC)

ACC	UNSW-NB15	CIC-IDS-2017
MLP	87.81	88.63
MLP + SSAL	98.37	98.82
CNN	91.44	90.52
CNN + SSAL	97.53	97.74

Table 5. Performance with different classifier.(ACC)

CIC-IDS-2017	SVM	k-NN
w/o SSAL	85.62	81.46
with SSAL	96.72	94.91

5 Conclusion and Discussions

In this paper, we try to tackles the data imbalance problem in network intrusion detection with adversarial style data augmentation and self-supervised contrastive representation learning. More specifically, we proposed a self-supervised adversarial learning way to enhance the representative learning progress in deep learning based NIDS, which utilizing a instance-wise attack to yield a robust model by suppressing theirs adversarial vulnerability against perturbation samples. State-of-the-art performance was achieved on commonly used Experiments on multiple datasets show improvement of proposed learning framework against vanilla DL approach with same backbones.

In addiction to the conclusion, there are also some works could be done in the future. Although we among other researchers have made a lot of effort on data imbalance for network intrusion detection problems, there are still more gaps need to be filled to a robust and applicable NIDS. For instance, in our method the result from different feature sets shows noticeable performance gap. we believe that to further improve the representative ability of network flow data with a standard and comprehensive behavior feature set is key to better data-driven NIDS solution. Also we are looking forward to explore an universal end-to-end approach for more generalized NIDS which could greatly reduces the difficulty of system deployment.

References

1. Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., Ahmad, F.: Network intrusion detection system: a systematic study of machine learning and deep learning approaches. Trans. Emerg. Telecommun. Technol. **32**(1), e4150 (2021)
2. Aouini, Z., Pekar, A.: Nfstream: a flexible network data analysis framework. Computer Networks, p. 108719 (2022)
3. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning, pp. 1597–1607. PMLR (2020)
4. Chou, D., Jiang, M.: A survey on data-driven network intrusion detection. ACM Comput. Surv. **54**(9), 1–36 (2021)
5. Chowdhury, M.M.U., Hammond, F., Konowicz, G., Xin, C., Wu, H., Li, J.: A few-shot deep learning approach for improved intrusion detection. In: 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), pp. 456–462. IEEE (2017)
6. Claise, B.: Cisco systems netflow services export version 9. RFC **3954**, 1–33 (2004)
7. Damasevicius, R., et al.: Litnet-2020: an annotated real-world network flow dataset for network intrusion detection. Electronics **9**(5), 800 (2020)
8. Ding, H., Chen, L., Dong, L., Fu, Z., Cui, X.: Imbalanced data classification: a KNN and generative adversarial networks-based hybrid approach for intrusion detection. Future Gener. Comput. Syst. **131**, 240–254 (2022)
9. Dokas, P., Ertoz, L., Kumar, V., Lazarevic, A., Srivastava, J., Tan, P.N.: Data mining for network intrusion detection. In: Proceedings of the NSF Workshop on Next Generation Data Mining, pp. 21–30. Citeseer (2002)
10. Ferrag, M.A., Maglaras, L., Moschoyiannis, S., Janicke, H.: Deep learning for cyber security intrusion detection: approaches, datasets, and comparative study. J. Inform. Secur. Appl. **50**, 102419 (2020)
11. Gao, X., Shan, C., Hu, C., Niu, Z., Liu, Z.: An adaptive ensemble machine learning model for intrusion detection. IEEE Access **7**, 82512–82521 (2019)
12. Gutmann, M., Hyvärinen, A.: Noise-contrastive estimation: a new estimation principle for unnormalized statistical models. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, pp. 297–304. JMLR Workshop and Conference Proceedings (2010)
13. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), vol. 2, pp. 1735–1742. IEEE (2006)
14. Hindy, H., et al.: Leveraging siamese networks for one-shot intrusion detection model. arXiv preprint arXiv:2006.15343 (2020)
15. Ho, C.H., Nvasconcelos, N.: Contrastive learning with adversarial examples. Adv. Neural Inform. Process. Syst. **33**, 17081–17093 (2020)
16. Jeong, H.D.J., Hyun, W., Lim, J., You, I.: Anomaly teletraffic intrusion detection systems on hadoop-based platforms: a survey of some problems and solutions. In: 2012 15th International Conference on Network-Based Information Systems, pp. 766–770. IEEE (2012)

17. Karatas, G., Demir, O., Sahingoz, O.K.: Increasing the performance of machine learning-based IDSS on an imbalanced and up-to-date dataset. IEEE Access **8**, 32150–32162 (2020)
18. Kdd cup 1999: Computer network intrusion detection (1999). http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
19. Keegan, N., Ji, S.-Y., Chaudhary, A., Concolato, C., Yu, B., Jeong, D.H.: A survey of cloud-based network intrusion detection analysis. Hum. Centric Comput. Inform. Sci. **6**(1), 1–16 (2016). https://doi.org/10.1186/s13673-016-0076-z
20. Lee, J., Park, K.: Gan-based imbalanced data intrusion detection system. Person. Ubiquitous Comput. **25**(1), 121–128 (2021)
21. Liu, C., et al.: Learning a few-shot embedding model with contrastive learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 8635–8643 (2021)
22. Liu, L., Wang, P., Ruan, J., Lin, J.: Conflow: contrast network flow improving class-imbalanced learning in network intrusion detection. Research Square Preprint (2022)
23. Manocchio, L.D., Layeghy, S., Portmann, M.: Flowgan-synthetic network flow generation using generative adversarial networks. In: 2021 IEEE 24th International Conference on Computational Science and Engineering (CSE), pp. 168–176. IEEE (2021)
24. Manzoor, M.A., Morgan, Y.: Real-time support vector machine based network intrusion detection system using apache storm. In: 2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), pp. 1–5. IEEE (2016)
25. McKeown, N., et al.: Openflow: enabling innovation in campus networks. ACM SIGCOMM Comput. Commun. Rev. **38**(2), 69–74 (2008)
26. Moustafa, N., Slay, J.: 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), pp. 1–6. IEEE (2015)
27. Van den Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv e-prints pp. arXiv-1807 (2018)
28. Pan, T., Zhao, J., Wu, W., Yang, J.: Learning imbalanced datasets based on smote and gaussian distribution. Inform. Sci. **512**, 1214–1233 (2020)
29. Sarhan, M., Layeghy, S., Portmann, M.: Towards a standard feature set for network intrusion detection system datasets. Mobile Networks Appl. **27**(1), 357–370 (2022)
30. Schultz, M., Joachims, T.: Learning a distance metric from relative comparisons. Adv. Neural Inform. Process. Syst. **16** (2003)
31. Sharafaldin, I., Gharib, A., Lashkari, A.H., Ghorbani, A.A.: Towards a reliable intrusion detection benchmark dataset. Softw. Network. **2018**(1), 177–200 (2018)
32. Shiravi, A., Shiravi, H., Tavallaee, M., Ghorbani, A.A.: Toward developing a systematic approach to generate benchmark datasets for intrusion detection. Comput. Secur. **31**(3), 357–374 (2012). https://doi.org/10.1016/j.cose.2011.12.012, https://www.sciencedirect.com/science/article/pii/S0167404811001672
33. Thomas, R., Pavithran, D.: A survey of intrusion detection models based on NSL-KDD data set. In: 2018 Fifth HCT Information Technology Trends (ITT), pp. 286–291 (2018)
34. Wang, T., Lv, Q., Hu, B., Sun, D.: A few-shot class-incremental learning approach for intrusion detection. In: 2021 International Conference on Computer Communications and Networks (ICCCN), pp. 1–8. IEEE (2021)

35. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3733–3742 (2018)
36. Xu, C., Shen, J., Du, X.: A method of few-shot network intrusion detection based on meta-learning framework. IEEE Trans. Inform. Foren. Secur. **15**, 3540–3552 (2020)
37. Yu, L., et al.: PBCNN: packet bytes-based convolutional neural network for network intrusion detection. Comput. Networks **194**, 108117 (2021)
38. Zhang, H., Huang, L., Wu, C.Q., Li, Z.: An effective convolutional neural network based on smote and gaussian mixture model for intrusion detection in imbalanced dataset. Comput. Netw. **177**, 107315 (2020)