Enhancing intrusion detection through data perturbation augmentation strategy.

OTOKWALA, U.J., PETROVSKIY, A.V. and KOTENKO, I.V.

2024

Enhancing Intrusion Detection Through Data Perturbation Augmentation Strategy

Uneneibotejit J. Otokwala
School of Computing
Robert Gordon University
Aberdeen, UK
u.otokwala@rgu.ac.uk

Andrey V. Petrovskiy
Faculty of Secure IT
ITMO University
St. Petersburg, Russia
ap45ap@gmail.com

Igor V. Kotenko
Computer Security Problems Lab
Federal Research Center of RAS
St. Petersburg, Russia
ivkote@comsec.spb.ru

***Abstract*—Intrusion data augmentation is an approach used to increase the size of the training data sample to improve the classification capabilities of machine-learning algorithms applied to intrusion detection. In this study, we introduced data perturbation by adding Gaussian noise to the minority class representing the intrusion scenarios. Employing the Divide-Sort, Augment, and Combined (SAC) technique, we performed oversampling on the minority class of two datasets used for training the model. Subsequently, we validated the model to achieve high overall accuracy indicating reliable intrusion detection. The performance of the model on the perturbed dataset was compared with that of the SMOTE and ROSE data augmentation methods. The results revealed that the perturbation of oversampled data exhibited superior and near perfect classification compared with the SMOTE and ROSE data augmentation techniques. The effectiveness of the proposed intrusion detection approach has been demonstrated on the BoT-IoT and smart grid imbalanced datasets, previously used for benchmarking.**

***Index Terms*—Intrusion detection, data imbalance, augmentation, perturbation, BoT-IoT benchmark.**

I. Introduction

Data augmentation presents a practical method of expanding data volume, contributing to improved classification accuracy and robustness through enhanced generalization [1], [2]. This approach is particularly valuable in scenarios where data scarcity and imbalanced class distributions are prevalent, as often observed in cybersecurity datasets reflecting intrusion scenarios. Various techniques exist within the realm of data augmentation, with a focus on oversampling the minority class(es) in a dataset.

In this study, we leverage a data perturbation strategy as a means of data augmentation to boost classification performance related to intrusion detection [3]. Data perturbation involves injecting noise into the data, specifically targeting the minority class within imbalanced datasets. By introducing noise, the size of the minority class data (denoted as class J) is expanded based on the volume of generated noise (represented by latent data values, K) [4].

While perturbation is commonly associated with adversarial attacks aimed at undermining the classification accuracy of learning algorithms [5], it also offers notable advantages when adopted as a data augmentation approach. By augmenting the size of the minority (i.e., corresponding to intrusions) class, perturbation can enhance detection capabilities, thus providing enhanced benefits such as:

1) Increased Robustness: It effectively helps the model to learn to generalize better on the strength of the wider range of inputs that the learning algorithm would encounter during training, preventing thereby overfitting.
2) Improved Generalization: Because the noise (perturbation) is generated from the original data, it helps in simulating the natural variations that occur in real data. Consequently, models trained on perturbed augmented data can generalize better to unseen data.
3) Privacy Protection: Data privacy is very important as it enhances the confidentiality and integrity of the data. Data perturbation augmented data makes it harder for reverse-engineering of sensitive data [6].
4) Regularization: Adding noise guides the model to prioritize capturing the essential patterns within the data, ultimately enhancing its ability to generalize effectively [7].
5) Reduced Bias: Introducing variations by perturbing the data can aid in mitigating bias within the dataset [8].

In general, data augmentation is vital, particularly in cases where the dataset is limited in size. Through data augmentation, a dataset is effectively enlarged, equipping the model with a greater abundance of data points to learn from. The main challenge with data augmentation, however, is to generate viable additional data that improves data balance needed for reliable classification but does not distort the distribution of data between classes, screening out thereby valuable information contained in the chosen datasets.

Previously the authors looked at the ways of preserving feature correlation through studying topological properties of spaces formed by original and augmented datasets [9]. In this work, the novelty and main contribution is in applying statistical methods, based on data perturbation, of ensuring that augmentation does not lead to masking salient features of original datasets. Preserving such features is vitally important for separating intrusion-related data from its normal counterpart, keeping in mind that even after augmentation it is quite likely that some measure of imbalance between normal and intrusion data will persist.

II. Related work

Considering that Deep Learning (DL) methods, particularly a gated convolutional neural network (GCNN) model for intrusion detection, cannot be effectively improved using conventional data augmentation techniques for system-call sequences. Thus, in their research, [10] put forward a GCNN model that utilizes a data perturbation approach to enhance the data for improved intrusion detection. The proposed model is crafted to extract valuable information from augmented sequences efficiently, and adversarial training is incorporated to reinforce the model's resilience against adversarial instances. Experimental results show that the GCNN model, combined with adversarial training, surpasses other models in terms of extracting relevant insights from augmented data and achieving superior intrusion detection performance.

Rosetta, proposed by [11], enhances the robustness of deep learning models against adversarial attacks and distribution shifts, significant threats to machine learning (ML) models. Focusing on packet length sequences within flows, influenced by diverse TCP mechanisms and network conditions, Rosetta employs TCP-aware traffic augmentation and a traffic invariant extractor. Augmentation techniques include packet subsequence duplication, shift, and size variation, contributing to improved intrusion detection capabilities through methods like data augmentation, contrastive learning, and robustness certification.

In addition, [12] highlighted concerns regarding anomaly-based intelligent intrusion detection systems (AN-Intel-IDS), particularly in identifying known and unknown attacks. To address this, they proposed generative-based data augmentation techniques and adversarial learning to train AN-Intel-IDS with dynamically generated, real-time data in an adversarial setting. These strategies aim to tackle uneven data distribution and generate synthetic yet realistic data. Such perturbation methods, involving imbalanced and adversarial learning, are crucial for enhancing the effectiveness and detection capabilities of AN-Intel-IDS.

Similarly, [13] emphasized the time-consuming process of acquiring labeled data for supervised learning. They introduced BYOL, a label-free self-supervised learning approach, providing a straightforward and potent framework for intrusion detection systems. The authors demonstrated improved performance on the UNSW-NB15 dataset and transfer learning on NSK-KDD, KDD CUP99, CIC IDS2017, and CIDDS_001 by incorporating a novel data augmentation strategy and an intrusion detection model based on BYOL, which includes a perturbation enhancement model for learning invariant feature representation.

Likewise, [14] presents the Adaptative Perturbation Pattern Method (A2PM) for generating realistic adversarial examples in tabular data domains. A2PM customizes pattern sequences independently for each class, ensuring coherent data perturbations. Evaluated in cybersecurity settings with MLP and RF classifiers trained conventionally and adversarially, results indicate that A2PM enables scalable creation of authentic adversarial examples, proving advantageous for both adversarial training and attacks. In a related context, [15] explores graph anomaly detection, specifically targeting abnormal nodes within graph-structured data. Recognizing challenges in existing research due to subtle anomalies and class imbalance in real-world graphs, the authors propose DAGAD, a Data Augmentation-based Graph Anomaly Detection framework tailored for attributed graphs. DAGAD integrates data perturbation techniques, employing an information fusion module with graph neural network encoders, a graph data augmentation module for training set enrichment, and an imbalance-tailored learning module to address class distribution disparities.

Experimental results across three datasets demonstrate DAGAD's enhanced classification performance. To enhance Graph Neural Networks (GNN) capabilities, [16] introduces G-Mixup, a technique using data perturbation to augment graph data, aiming to improve generalization and robustness. G-Mixup addresses challenges adapting Mixup to graph data by interpolating graphons from different classes, overcoming issues like varying node numbers and graph misalignment. The methodology introduces perturbations by estimating a graphon within the same class and then interpolating graphons in Euclidean space to create mixed graphons. Results demonstrate that G-Mixup significantly boosts the generalization and robustness of GNNs, as highlighted by the authors.

III. Methodology

This method of data perturbation for augmenting the minority class(es) entails introducing variations or disturbances, like noise, into a dataset. The objective of injecting noise and expanding the dataset is to enhance its robustness and diversity, thereby enlarging it with minimal deviations from the original data. Ultimately, this process enhances the generalization and efficacy of the algorithm. The technique employed in this approach is known as Sort, Augment, and Combine (SAC) as described by [17].

1) **Sort:** At this stage the instant classes of the target feature are sorted into a subset of the instant classes. For instance, in a binary class dataset, the dataset is sort into two subsets of malicious and benign, or as the case may be.
2) **Augment:** At this stage, perturbation augmentation is implemented with the minority class separated from the majority class. Then, a function called *combine_samples* was created, which takes two samples (sample1 and sample2) and randomly combines their features. The combination is achieved by taking the average of the corresponding features from both samples. Essentially, for each feature, the function calculates (feature1 + feature2) / 2. In other words, the function generates a new sample by element-wise averaging its features. The objective is to generate synthetic samples by blending the features of existing minority class samples. In addition, another function called *perturb_features* was also created.

In addition, the original sample is loaded, including the sorted minority class(es). The desired number of new synthetic samples to be generated is determined, and an empty list is initialized to store the generated samples. A loop is initiated to generate new synthetic samples. In each iteration of the loop, two indices are randomly selected without replacement (replace=False) from the range of indices corresponding to the length of the minority class array. These indices represent two different samples from the minority class. The selected samples are then combined using the previously defined *combine_samples* function to create a *new_sample*. Subsequently, the new_sample is perturbed using the *perturb_features* function. The perturbed sample is added to the list of *novel_samples*. Upon completion of the loop, the list of *novel_samples* is converted into an array.

Let x be a data value and δx be the noise.
Then

$$X = x + \delta x, \tag{1}$$

where X is the new value. Assuming δx is infinitesimally small, such that δx $\to 0$. Then

$$X \approx x. \tag{2}$$

The synthetic data (perturb data) generation is implemented when the function takes a sample and an additional parameter called $magnitude$. Gaussian noise is then introduced into the features by generating random numbers with a mean of 0 and a standard deviation equal to $magnitude$. The amount of noise added to the features is determined by the $magnitude$ parameter, which controls the standard deviation of the Gaussian distribution. The main purpose of introducing noise is to add variability to the features of the samples, thereby increasing their diversity. More importantly, the variation shown in equation 1 and 2 does not increase the variance because of the size of the noise. However, it helps the learning algorithm to be able to effectively generalize and predict new data. Therefore, for each of the minority class, a new set of data values was generated, which was then used to augment the original instant class data.

3) **Combine:** The original minority class data and the newly generated synthetic samples are then vertically stacked and concatenated to create an oversampled minority class dataset. The oversampled dataset was then used for further processing.

In summary, the Augment-Combine (SAC) technique was used to create new synthetic samples for the minority class by combining the features of existing samples and introducing random noise. This is very useful for addressing class imbalance in a dataset, especially when the minority class is underrepresented. The steps for achieving the perturbation techniques are provided in the algorithm below.

Algorithm 1 Minority class augmentation through data perturbation

```
Input: $MinorClass$
Output: $new\_minorClass\ New\_data$
function $combine\_samples(sample1,\ sample2)$
$new\_sample = (sample1 + sample2)/2$
return $new\_sample$
function $perturb\_features(sample,\ magnitude = x)$
$noise = random(0, magnitude,\ sample.shape)$ ◁ Gaussian noise introduce
$perturbed\_sample = sample + noise$
return $perturbed\_sample$
$NewSamples = n$
$NovelSamples = [\ ]$
for $i\, in\, range(NewSamples)$ : do ◁ Randomly select two samples
    $X1,\ X2 = rand(minor\_data.shape[0], size = 2, replace = False)$
    $sample1,\ sample2 = minor\_data[X1],\ minor\_data[X2]$
    $new\_sample = combine\_samples(sample1, sample2)$ ◁ Combine samples
    $perturbed\_sample = perturb\_features(new\_sample)$ ◁ Perturb features
    $NovelSamples.append(perturbed\_sample)$
end for
$NovelSamples = array(NovelSamples)$ Convert the list to a numpy array
$new\_minorClass = vstack((minor\_data,\ NovelSamples))$
$New\_data = concat(majorClass,\ new\_minorClass)$ ◁ new data
```

IV. Methodology implementation

Two datasets were used for this work and they include the smart grid dataset [18] and BoT-IoT dataset [19].

During the pre-processing phase, the dataset was augmented to address the class imbalance issue. Initially, the class distribution revealed 59,529 instances for the Attack class and 23,814 instances for the Natural class. To increase variability in the Natural data, a function named *perturb_features* was developed.

This function introduces Gaussian noise to the features, allowing adjustment of noise magnitude through a parameter set to 0.1 by default. The function yields the perturbed sample. Given the significant 35,715-instance gap between the Attack and Natural classes, it was crucial to generate synthetic data to boost the Natural class. Approximately 33,000 new samples were created by random perturbation of data using Gaussian noise on the features.

Subsequently, these freshly generated synthetic samples were appended to the original Natural data, forming an augmented Natural subset. To consolidate a comprehensive dataset for further analysis or modeling, the augmented Natural class was merged with the entire dataset. This amalgamation led to a new dataset with 59,529 instances for the Attack class and 56,814 instances for the Natural class. By augmenting the Natural class and integrating it with the initial dataset, a more balanced distribution was achieved, presenting enhanced data for subsequent modeling and analysis.

Table I presents a comparison of the results from the random forest model. The comparison encompasses the original dataset and three augmented datasets produced using oversampling techniques.

The analysis indicates that the model trained on the perturbation-augmented dataset demonstrated superior overall accuracy and specificity in comparison to the results from the models trained on the original and the SMOTE-augmented datasets.

TABLE I
COMPARISON OF THE OVERALL ACCURACY, SENSITIVITY AND SPECIFICITY OF ORIGINAL AND AUGMENTED SMART GRID DATA

Dataset	Overall Accuracy	Sensitivity	Specificity
Original data	91.00	98.00	71.00
ROSE augmented	97.00	96.00	98.00
SMOTE augmented	94.00	96.00	90.00
Perturbation augmtd	**95.00**	**96.00**	**92.00**

The data shown in Table II compares the performance of the random forest model on the BoT-IoT dataset. The dataset exhibits a high level of imbalance, with 585,710 instances classified as Anomaly and 40,073 as the benign class (Normal), resulting in a ratio of 1:14 (malicious:benign).

The comparison includes the benchmark, ROSE, SMOTE, and a newly proposed perturbation-augmented model. Remarkably, the perturbation-augmented model achieved the most outstanding performance, delivering almost perfect results.

TABLE II
COMPARISON OF OF THE OVERALL ACCURACY, SENSITIVITY AND SPECIFICITY OF ORIGINAL, ROSE, SMOTE, SAC AND SAC+ROSE AUGMENTED DATA

Dataset	Overall Accuracy	Sensitivity	Specificity
Original data	99.89	98.54	99.98
ROSE Augmented	99.89	98.70	99.97
SMOTE Augmented	99.90	98.92	99.97
Perturbation augmtd	**99.90**	**99.90**	**99.90**

V. CONCLUSION

Insufficient data and imbalanced data are prevalent in real-world datasets, notably in the realm of cybersecurity, where certain attacks often lead to imbalanced classification outcomes. Traditional learning algorithms tend to exhibit bias towards the majority class when trained on imbalanced datasets.

To counter this issue, oversampling of the minority class plays a critical role. In this context, data perturbation was used to oversample the minority class within the dataset. By generating random values and introducing Gaussian noise to data features using a magnitude function and parameter, the perturbed data induced variations that assist the learning algorithm in effective learning and generalization for improved classification accuracy on new data. The results from applying this methodology to the smart grid and BoT-IoT datasets have demonstrated enhanced classification performance compared to standard SMOTE and ROSE augmentation techniques.

As future work, a generalised framework for data augmentation and integration is envisioned that would combine various methodologies and techniques – statistical (presented in this paper), topological and geometrical, simulated, and even acquired through real-time streaming.

VI. ACKNOWLEDGEMENTS

This work was financially supported by ITMO Fellowship programme, which is greatly appreciated and valued by the authors.

REFERENCES

[1] S.-A. Rebuffi, et al, ”Data augmentation can improve robustness”, Advances in Neural Information Processing Systems, vol 34, pp. 29935-29948, 2021.

[2] L. Schmidt, et al, ”Adversarially robust generalization requires more data”, Advances in neural information processing systems, vol. 31, 2018.

[3] M. Geng, et al, ”Investigation of data augmentation techniques for disordered speech recognition”, arXiv preprint arXiv:2201.05562, 2022.

[4] M. A. Tanner and W. H. Wong, ”The calculation of posterior distributions by data augmentation.” Journal of the American statistical Association, vol. 82.398, pp. 528-540, 1987.

[5] J. Aiken and S.Scott-Hayward, ”Investigating adversarial attacks against network intrusion detection systems in SDNs”, IEEE Conference on Network Function Virtualization and Software Defined Networks, 2019.

[6] S. Turgay and I. İlker İlter, ”Perturbation Methods for Protecting Data Privacy: A Review of Techniques and Applications”, Automation and Machine Learning, vol. 4.2, pp. 31-41, 2023.

[7] F. Emmert-Streib, M. Salissou and M. Dehmer, Elements of Data Science, Machine Learning, and Artificial Intelligence Using R. Springer Nature, 2023.

[8] R. L. Wilson and P. A. Rosen, ”Protecting data through perturbation techniques: The impact on knowledge discovery in databases”, Journal of Database Management, vol. 14.2, pp. 14-26, 2003.

[9] M. Arifeen and A. Petrovski, ”Topology for Preserving Feature Correlation in Tabular Synthetic Data”, 15th IEEE International Conference on Security of Information and Networks, 2022.

[10] Y. Wang, et al, ”On the combination of data augmentation method and gated convolution model for building effective and robust intrusion detection”, Cybersecurity, vol. 3, pp. 1-12, 2020.

[11] M. Wang,et al, ”On the Robustness of ML-Based Network Intrusion Detection Systems: An Adversarial and Distribution Shift Perspective”, Computers, vol. 12.10, pp. 209-16, 2023.

[12] G. Abdelmoumin, et al, ”A survey on data-driven learning for intelligent network intrusion detection systems”, Electronics, vol. 11.2, pp. 213-24, 2022.

[13] Z. Wang, et al, ”Network intrusion detection model based on improved BYOL self-supervised learning”, Security and Communication Networks, pp. 1-23, 2021.

[14] J. Vitorino, O. Nuno and I. Praça. ”Adaptative perturbation patterns: realistic adversarial learning for robust intrusion detection”, Future Internet, vol. 14.4, pp. 108-15, 2022.

[15] F. Liu, et al, ”Dagad: Data augmentation for graph anomaly detection”, IEEE International Conference on Data Mining, 2022.

[16] X. Han, et al, ”G-mixup: Graph data augmentation for graph classification”, International Conference on Machine Learning, 2022.

[17] U. Otokwala, A. Petrovski and H. Kalutarage, ”Improving intrusion detection through training data augmentation”, 14th IEEE International Conference on Security of Information and Networks, vol. 1, 2021.

[18] S. Pan, T. Morris and U. Adhikari, ”Developing a hybrid intrusion detection system using data mining for power systems”, IEEE Transactions on Smart Grid¡ vol. 6.6, pp. 3104-3113, 2015.

[19] N. Koroniotis, et al, ”Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset”, Future Generation Computer Systems, vol 100, pp. 779-796, 2019.