International Journal of Cloud Computing and Cyber Security (IJCCCS)
ISSN: XXXX-XXXX / Volume 1 Issue 2 November-December 2026
Paper Id: JCCCS -V1I2P104/ Page No: 44-58
Doi: XX.XXXXX/XXXXXXX/ IJCCCS -V1I2P104

Original Article

Measuring the Impact of Synthetic Training Data on Cloud Intrusion Detection under Class Imbalance

Wenke Lee
Georgia Institute of Technology.

Received Date: 16 November 2026 Revised Date: 20 November 2026 Accepted Date: 23 November 2026

Abstract: *The threat landscape for cloud computing environments continues to evolve and grow more complex. Threats such as unauthorized access, distributed denial-of-service (DDoS) attacks, credential theft, malware activity, and moving laterally through virtualized infrastructure present significant challenges to the security of these systems. Intrusion detection systems (IDSs) provide one means of detecting and preventing these types of malicious actions. However, the performance of IDSs is often limited due to "class imbalance" issues within the data sets used to train them. Class imbalance refers to situations in which the majority class (e.g., normal or benign network traffic) greatly outnumbers the minority class (e.g., intrusions). When machine learning models are trained with imbalanced data, they tend to predict the majority class and fail to recognize rare, yet highly damaging attacks. One way to address the issue of class imbalance is through the use of synthetic training data. Synthetic training data involves creating additional minority-class examples to assist machine learning models in learning more effectively from the data set. In this research, we investigate the effects of using synthetic training data on cloud intrusion detection under various levels of class imbalance. We have developed an experimental framework that allows us to compare conventional training, random oversampling, and synthetic sampling approaches. Our evaluation metrics include precision, recall, F1-score, area under the receiver operating characteristic curve (AUC-ROC), and area under the precision-recall curve (AUC-PR). Additionally, we evaluate the false-positive rates of each approach, the computational overhead involved in implementation, and the ability of detection models to generalize to previously unseen attack patterns. We place particular emphasis on preventing data leakage, accurately representing real-world traffic, and understanding the limitations of artificially generated attack samples. Rather than assuming that increasing synthetic data will automatically enhance security, we assess whether adding synthetic data results in quantifiable and practical improvements. Our proposed investigation offers a methodical approach to determining which data-balancing strategy best supports reliable, efficient, and robust intrusion detection in dynamic cloud environments.*

Keywords: *Cloud Security, Intrusion Detection, Synthetic Training Data, Class Imbalance, Machine Learning, Anomaly Detection, Data Augmentation, Cybersecurity, Minority-Class Detection, Model Evaluation.*

I. INTRODUCTION

In today's digital world, cloud computing is an integral part of the overall digital infrastructure. Cloud computing provides organizations with flexible access to a wide range of computing resources such as scalable computing power, distributed data storage, virtualized machines, container-based applications, and networked services, all available when needed. Organizations have taken advantage of cloud computing's flexibility and affordability to implement cloud computing across many sectors including healthcare, finance, education, manufacturing, government, and IT. However, the decentralized and interconnected nature of cloud computing creates numerous cybersecurity risks. Cloud-based systems can be vulnerable to unauthorized entry into the system; DDoS attacks that disrupt normal operations; malware infections; credential theft; exfiltration of sensitive information; and exploitation of vulnerabilities within virtualized resources. To help mitigate some of these risks, organizations can utilize Intrusion Detection Systems (IDS). IDS enable organizations to monitor network activity; detect potentially harmful or suspicious behaviors; and respond quickly to potential security incidents. Additionally, machine learning has enhanced IDS capabilities by allowing organizations to identify patterns among network activity and potential threats, without having to rely solely upon previously established attack signatures.

Although machine learning can greatly enhance IDS performance, its effectiveness relies heavily on the quality, diversity, and representative nature of its training data. A significant problem that remains today is class imbalance. Class imbalance occurs when the amount of normal network activity far exceeds the amount of malicious activity. This condition can lead to conventional machine learning algorithms being biased toward the majority class (normal activity), and therefore report falsely

This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/2.0/)

high accuracy levels. For organizations utilizing cloud-based systems, this condition is particularly problematic since even a relatively small number of successful attacks against a cloud-based system could result in the loss of proprietary information; loss of critical services; or create opportunities for additional malicious actions.

Several existing reviews of the literature have also examined resampling techniques; SMOTE (Synthetic Minority Over-sampling Technique); and Generative Adversarial Networks (GANs) as possible methods for mitigating issues related to class imbalance in intrusion detection datasets. Synthetic training data is becoming an alternative to help represent less represented attack categories in machine learning datasets. Instead of just copying the minority-class examples, synthetic data generates more examples by utilizing statistical interpolation, learned probability distribution, or generative models. For example, SMOTE generates new samples that fall within the space of the two nearest neighbors of the same minority class. CGANs and CTGANs can be used to generate new samples that are based upon the patterns found within the training data, and then conditioned on a specific class. Synthetic data can provide a wider variety of attack representations, create better decision boundaries for classifiers, and allow classifiers to recognize rare intrusion patterns. Recently, research has explored the use of synthetic data generation to create more balanced cybersecurity datasets, and thus increase the ability of classifiers to identify malicious network traffic.

Synthetic augmentation's effectiveness cannot be assumed to remain constant across all algorithms, datasets, or types of attacks. Synthetic samples can reproduce biases present within the real world; they can produce artificial combinations of features that do not exist in reality; and they can fail to represent the temporal and behavioral characteristics of true attacks. Additionally, generative models may suffer from "mode collapse," which occurs when the model produces only a small subset of possible patterns repeatedly, thereby limiting the diversity of the augmented dataset. Moreover, overly aggressive augmentation can alter the original distribution of network traffic, causing the model to over-fit on artificially created characteristics instead of learning transferrable characteristics of malicious behavior. All these issues are particularly relevant in cloud environments where network traffic is highly variable across tenants, applications, workloads, and deployment architectures. A model trained on synthetic samples that are very similar to those found in one specific dataset may not generalize well to traffic collected under different conditions. Finally, another major issue is data leakage: synthetic samples must be generated solely based on training data, after the dataset has been divided appropriately, so that no information from evaluation samples influences model development. Therefore, evaluating synthetic training data involves more than simply evaluating the overall accuracy of a classifier. To truly evaluate whether synthetic training data provides practical benefits rather than just producing good results for a single set of experimental parameters, we must evaluate how it affects minority-class recall, precision, F1-score, false positive rates, precision-recall curve performance, computational overhead, and generalization to previously unseen observations. We must also determine if there are differences in performance across various imbalance ratios and generation techniques.

The primary focus of this study, titled "Assessing the Impact of Synthetic Training Data on Cloud Intrusion Detection with Class Imbalances," is to establish a systematic methodology for assessing the effects of synthetic training data on the performance of machine-learning-based intrusion detection. A primary goal is to evaluate the differences between models developed using naturally imbalanced datasets and models developed using alternative balancing methods (e.g., random oversampling, SMOTE) or, when computationally feasible and appropriate for the dataset's characteristics, generative approaches. This evaluation can utilize existing public intrusion detection datasets that include labeled benign and malicious network traffic, but will also document the suitability of those datasets as representations of cloud-related security scenarios. Datasets selected for inclusion in this study should be chosen based on their diversity of attack types, availability of features, class distributions, and data collection methodologies, rather than assuming that each dataset represents all types of cloud deployments. The evaluation process should involve cleaning the data, preprocessing features, partitioning the dataset either via stratification or by time-aware partitioning, augmenting only the training subset of the partitioned data, developing the models, and conducting independent testing.

Traditional classification models (e.g., random forests, gradient boosted machines) can be utilized as baseline comparisons, but more complex models can be added to the comparative analysis. Performance can be reported using macro-averaged F1-scores, per-class precisions and recalls, areas under precision-recall curves, and false positive rates; in addition to traditional measures such as accuracy and computational resource utilization. Finally, the evaluation should examine the similarity and diversity of generated samples and assess whether observed performance improvements are consistent across different levels of imbalance and random seed initializations. The primary questions of interest in this research include: (1) Does synthetic training data increase the ability of models to identify rare attacks? (2) Which generation strategies produce the most consistent

improvements in model performance? and (3) Are the performance improvements produced by synthetic training data sufficient to justify the additional computational expense associated with generating synthetic training data and potential degradation of data quality? This research is expected to contribute to a structured assessment methodology for determining the degree to which synthetic training data produces genuine increases in detection capabilities, as opposed to artifactual increases resulting from synthetic sampling techniques. By addressing issues related to experimental reproducibility, class-specific performance, data leakage, and generalizability, this research is designed to assist researchers and cloud security practitioners in making informed decisions about whether to adopt synthetic training data for cloud intrusion detection. Results presented herein should be interpreted relative to experiments actually conducted; if experimental testing has not yet been performed, then the proposed methodology represents research that is yet to be completed, rather than evidence of demonstrated performance improvements. Ultimately, this methodology can enable security researchers and cloud security professionals to determine when synthetic training data is advantageous, when existing balancing techniques are sufficient, and what safeguards are necessary to develop effective intrusion detection systems that address evolving cybersecurity threats.

II. RESEARCH METHODOLOGY FOR SYNTHETIC DATA GENERATION AND CLOUD INTRUSION DETECTION

The research design used to assess the effects of synthetic training data on cloud intrusion detection in the presence of class imbalance involves designing a structured experimental design that will evaluate various methods of addressing class imbalance and how they affect classification performance. The first phase of this study will involve selecting a valid intrusion detection dataset that contains both legitimate network traffic and several categories of malicious activity (e.g., denial of service attacks; unauthorized access attempts; scanning; and other security-related events). This chosen dataset must contain enough information about network connections, traffic properties, protocol behavior, packet statistics, and label information related to the malicious activities. After obtaining the dataset and prior to beginning model development, the dataset will undergo pre-processing to eliminate duplicates; handle missing values; resolve inconsistencies in labels; convert categorical variables into machine-readable forms; and standardize numerical variables as needed for the selected machine learning algorithms. Furthermore, irrelevant or redundant variables can be identified via feature selection techniques. After the dataset has been processed, it will then be divided into training, validation, and testing sets via a suitable method of division. Dividing the dataset into three distinct sets is critical since the testing set must be kept separate from the synthetic data creation procedure to allow us to estimate the model's ability to perform on novel observations.

The second phase evaluates the impact of varying types of synthetic data generation and balancing methods on minority-class representation. A baseline dataset maintains the original class distribution so that researchers can assess how well intrusion detection models perform without any additional balancing. Random oversampling is utilized to evaluate if creating copies of minority-class samples has an observable effect on performance. The Synthetic Minority Over-sampling Technique (SMOTE) creates new minority-class examples through interpolation among nearby minority-class examples; therefore, potentially enhancing the variability of training samples. When appropriate, generative models can be evaluated to create synthetic traffic records which mirror the characteristics of certain attack categories. All methods must only be applied to the training set, with validation data used for model selection and parameter tuning. The volume of synthetic data generated can be adjusted to determine if a smaller amount of augmentation leads to superior results compared to larger amounts of oversampling. Additionally, varying levels of class imbalance should be assessed to determine if the effectiveness of synthetic data varies as the frequency of malicious observations decreases. Synthetic records must be evaluated for statistical consistency, validity of features, accuracy of class-labels, and potentially unrealistic combinations of network characteristics. This phase will allow for a comparative evaluation of natural training distributions against various alternative augmentation strategies while reducing the risk of distortion due to data leakage or artificial sample duplication.

In stage three, I will evaluate each of the evaluated data sets with machine learning classifiers (decision tree, random forest, gradient boosted trees, support vector machine if possible). Each classifier should be trained in a similar manner so we can say that any difference in the performance of each classifier is due to the balancing strategy and not due to having different parameters set up. We should look at minority-class precision, recall, F1 score, macro average performance, and the area under the Precision-Recall Curve. Although accuracy and the Area Under the Receiver Operating Characteristic Curve (AUROC) may give some other insight into what is going on, I don't believe that these metrics should be looked at alone as large imbalances in the data can hide very bad detection of rare attack events. Also, false positive rate is important as too many false alarms will put more work onto Cloud Security Teams. We need to record how long it takes to train, run, memory usage, inference latency for each classifier so we know if it is commercially viable. Running multiple trials of experiments with different random seeds will allow us to see if the improvements we are seeing are consistent across all trials or if they are only present when the data is split

in a specific way. If possible, I would like to do some statistical analysis to determine if the differences in performance are significant enough to be more than just random variation.

The last step of the study is to evaluate the experimental results based on the effectiveness of security, the quality of the data and what would be required to implement the model into production. Synthetic training data is only advantageous if it improves the identification of attack categories that are not represented in the initial training data, and does so with acceptable levels of false positive rates and processing times. This portion of the evaluation should determine if some attack categories are improved more than others by using synthetic training data, and if synthetic training data can improve the model's ability to generalize to new types of network traffic that have never been seen before. There could potentially be several limitations to this research. Some of these may include the age of the dataset, incomplete representation of all possible types of attacks, synthetic records that do not accurately represent real world data, and differences between the experimental datasets and the actual operational environments where the model will be deployed. To ensure reproducibility of the research findings, the research must document the experimental design, preprocessing techniques used, models developed, evaluation criteria applied and how the data was partitioned. In general, the methodology used in this research creates a framework for evaluating when synthetic training data can enhance cloud intrusion detection systems and when alternative methods for addressing class imbalance should be utilized.

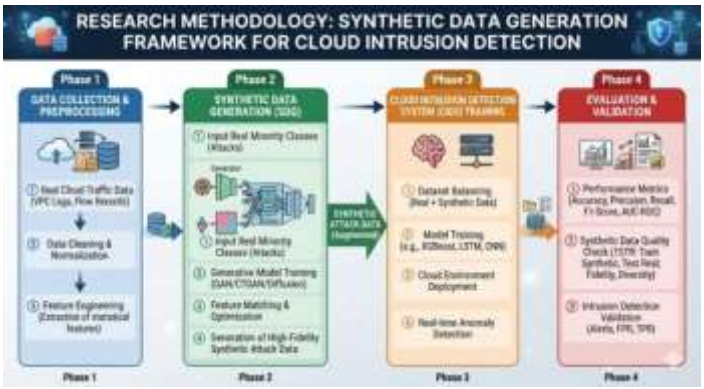


Figure 1: Proposed Research Methodology Framework for Synthetic Data Generation and Cloud Intrusion Detection

III. SYNTHETIC TRAINING DATA AND CLOUD INTRUSION DETECTION

A. Characteristics of Class Imbalance in Cloud Security

Class imbalance poses a major issue for cloud intrusion detection due to the fact that most legitimate connections occur much more frequently than those that are malicious. As such, most traffic within a cloud environment will consist of normal application requests, data transfers, authentication events, and service-to-service communication, all of which will produce vast amounts of benign traffic. On the other hand, the number of instances where an attacker would perform a specific action (such as privilege escalation, credential misuse, etc.) will typically be extremely small in comparison to the total amount of traffic observed. Therefore, machine learning algorithms developed using such highly skewed distributions can be heavily biased towards predicting the majority class. While this bias can result in high overall accuracy, it does not translate into good performance at identifying the few instances of attack. Furthermore, the issue of class imbalance is especially critical when there is a risk of severe consequences from failing to identify a malicious event. Additionally, the representation of minority classes can vary greatly among different types of cloud platforms, workloads, and attack categories, thereby making it challenging to design a universally applicable detection model. Thus, understanding both the magnitude and nature of class imbalance is crucial prior to generating synthetic data. Researchers need to assess class frequency, attack diversity, feature distribution, and the relationships among intersecting classes in order to determine if data augmentation is required and which minority categories need to be represented more fully.

B. Synthetic Data Generation Techniques for Intrusion Detection

Synthetic Data Generation: This process is used to create new training data for underrepresented attacks. New samples will be created from existing samples that represent similar characteristics of the attack being studied. Each technique has its own advantages and disadvantages. For example, random oversampling creates multiple copies of each minority-class sample; therefore, this method can lead to overfitting since there are no new samples provided. SMOTE uses interpolation to create new samples based upon the distance between two or more minority-class samples. It can extend the decision boundary of classifiers.

More advanced techniques include Conditional Generative Adversarial Networks (CGANs) and Tabular Generative Models (TGMs) which can learn complex interactions between network features and generate new samples with similar statistical properties. However, when generating synthetic data, the generated data must maintain the relationship between traffic characteristics, protocol behavior, and attack labels. Generating synthetic data incorrectly can lead to unrealistic network conditions or hinder the ability to distinguish benign and malicious activity. Thus, when choosing a generation technique, you should take into account the size of your dataset, the types of features you have, your available computing power, the degree of variability within the minority-class, and the specific detection task at hand. Evaluation of the different techniques is required to determine which technique would be most appropriate for your application.

C. Evaluation Metrics and Experimental Performance

In order to evaluate synthetic training data, researchers must use several different performance measures. Accuracy alone does not provide sufficient information about the ability of an intrusion detection system to detect attacks in the presence of imbalanced classes. Accuracy is defined as the percentage of correctly classified instances. However, if a classifier primarily classifies instances as benign, it could receive a high accuracy score even though it misses a significant number of attacks. On the other hand, precision is the percentage of instances classified as intrusive that were truly intrusive. Recall is the percentage of truly intrusive instances that were correctly classified. The F1-score combines precision and recall so that researchers can see the trade-off between identifying attacks and generating false positive alarms. Macro-averaged F1-score treats all classes equally; therefore, this metric is especially helpful when there are very large differences in the frequency of various types of attacks. In addition to measuring precision and recall, researchers may also find that assessing the area under the precision-recall curve will help them gain a better understanding of how well their models perform in terms of detecting minority-class instances. Finally, false positive rates indicate how often legitimate traffic is incorrectly identified as malicious. Researchers should also measure other operational characteristics such as training time, memory usage, and inference latency in order to evaluate operational efficiencies. Researchers will then be able to compare these performance measures across baseline, over-sampled, and synthetically augmented datasets to determine whether augmentation improves detection consistency or creates trade-offs.

D. Generalization, Reliability, and Practical Cloud Deployment

The real-world benefit of using synthetic training data will depend on the extent to which the improved classification results demonstrated in studies are consistent in a cloud environment that has varying levels of traffic, application use, user behavior, and attacks. Because cloud environments are continually undergoing changes in all of these areas, a model trained using one specific set of parameters may not work well with other types of work loads or previously unidentified threats. Therefore, only synthetic samples should come from the training data partition; validation and testing data need to remain separate to avoid any potential for data leak. Researchers need to determine if models developed with augmentation continue to function as expected across various random seed selections, imbalance ratios, and if possible, across other datasets or temporal periods. Additionally, researchers must evaluate if synthetic records maintain the relationship among features rather than just maintaining statistical similarities to the training data. At the point of deployment, researchers must find a balance between classification accuracy, processing overhead, number of alerts produced, scalability of the system, and time needed to respond to an event. Model evaluation at regular intervals and controlled retraining of models can assist in addressing changing attack methods.

IV. COMPARATIVE ANALYSIS OF SYNTHETIC DATA STRATEGIES FOR CLOUD INTRUSION DETECTION

The use of synthetic training data offers the potential to enhance cloud intrusion detection through improved representation of underrepresented attack types in highly unbalanced datasets. However, which strategy works best will depend upon several factors including; the characteristics of the initial traffic, the level of class imbalance, and the machine learning algorithm being utilized. To determine the advantages and disadvantages associated with various balancing methods and to provide recommendations regarding the selection of appropriate methodologies for real-world application of security technologies, a comparative study is needed. The comparative study must include evaluation criteria such as detection performance, minority-class representation, risk of overfitting, computational costs, and generalizability to future network traffic. Evaluation of the baseline training results will serve as a basis for comparison to assess improvements in performance. Random Oversampling, SMOTE (Synthetic Minority Over-sampling Technique), and Generative Models are examples of other methods used to modify the training distributions. All methods are to be assessed against the same partitioned datasets and with consistent performance metrics. The following table summarizes the anticipated characteristics of each approach and its potential contribution to the proposed research. The characteristics are qualitative estimates rather than experimentally derived measurements.

Table 1: Comparison of Data Balancing Strategies for Imbalanced Classification

Data strategy	Minority-class representation	Main advantage	Main limitation	Evaluation priority
Original imbalanced data	Preserves the original distribution	Provides a realistic baseline	Rare attacks may be overlooked	Recall and macro-F1
Random oversampling	Duplicates existing minority samples	Simple and computationally inexpensive	May increase overfitting	F1-score and generalization
SMOTE	Generates interpolated minority samples	Introduces additional feature combinations	May create unrealistic samples	Precision, recall, and sample validity
Conditional GAN	Generates class-conditioned synthetic records	Can model complex feature relationships	Training instability and mode collapse	Sample diversity and detection performance
Tabular generative models	Learns statistical relationships in structured data	Supports complex tabular feature distributions	Computational overhead and imperfect fidelity	Precision-recall performance and efficiency
Cost-sensitive learning	Gives greater importance to minority-class errors	Does not require synthetic samples	Requires suitable class weights or cost settings	Minority recall and false-positive rate
Hybrid balancing	Combines resampling and model-level adjustments	Can address multiple sources of imbalance	Increased complexity and tuning requirements	Robustness and computational cost

The comparative framework shows us that there isn't one best strategy that will result in the greatest outcomes in all cloud intrusion detection scenarios. Original imbalanced data is still needed as a reference point so we can see how a model performs with its own natural class distributions. Random oversampling is a simple way to increase minority class exposure; however, the repetition of records can lead to a model memorizing patterns, rather than developing the ability to recognize new attacks. SMOTE allows us to add variety, though interpolation might not work well with some types of categorical features, or complex relationships among variables. Generative approaches can give us many more varieties of examples if they are used properly. However, how well they perform will depend upon the quality of the data being used, how stable the models being generated are, and how realistic the generated observations are. Cost-sensitive learning gives us another option to compare against. This approach modifies how we assign value to classification errors without having to expand our data set artificially. Hybrid approaches may allow us to use combinations of all of these approaches, but we need to justify the increased complexity of such approaches with consistent experimental evidence.

For evaluation, the study must train each of the selected classifiers separately based on the strategies being evaluated, then evaluate the performance of the classifiers using the same untouched test set. Precision, recall, macro-F1, area under precision-recall curve, and false positive rate should be calculated for all relevant attack categories. In addition to measuring performance metrics, the study should also record computational resources used during the evaluations to assess if any identified performance improvements would be practical in a cloud environment with high resource demands and low latency requirements. Experiments may be repeated to evaluate if performance differences remain consistent across multiple runs, and statistical tests can be conducted to evaluate if performance differences are statistically significant. Finally, the study should assess quality of generated samples, diversity within classes, and potential for data leakage. This is necessary so that any identified performance improvements are not attributed solely to duplicate records or information being leaked from the test set to the training set.

V. ANALYSIS OF SYNTHETIC TRAINING DATA IN CLOUD INTRUSION DETECTION

A. Impact of Synthetic Data on Minority-Class Detection

synthetic training data could impact cloud intrusion detection systems due to how it impacts the number of training examples available to the machine learning algorithms. When there is an imbalance in the types of classes present in a dataset, the benign traffic generally accounts for the majority of the observations. On the other hand, the malicious activity (i.e., credential attacks, scanning, denial of service attempts, and unauthorized access) typically accounts for a smaller percentage of the total observations. Due to the imbalance of the classes, the classifier will tend to predict the majority class. Therefore, the classifier's ability to identify less frequent attacks is reduced. Synthetic data generation attempts to address the issue of class imbalance by generating more observations of the minority classes and by allowing the classification models to learn relevant patterns from these additional observations. There are several ways to generate synthetic data, including techniques like SMOTE which generates interpolated observations and generative models which attempt to reproduce more complex relationships between network traffic features.

Synthetic samples that accurately reflect the malicious activity of the attack can enhance minority class recall, strengthen decision boundaries, and enhance macro-averaged F1-scores; however, these benefits do not necessarily occur. Synthetic samples without realistic relationships among features can produce noise in the training set, decrease the separation between the benign and malicious classes, or cause models to identify artificial patterns. In addition, too much augmentation can change the training set distribution such that false positive rates will increase when the model is exposed to legitimate traffic. Thus, the effects of synthetic data should be studied across multiple types of attacks and imbalance levels rather than just evaluating based upon overall accuracy. Evaluations should include precision, recall, F1-scores, and the areas under precision-recall curves, specifically for those attacks that are hard to recognize due to few training examples available. Evaluations should also assess how well synthetic data compares to random oversampling and cost-sensitive learning to see if improvements in performance are due to increased representation of the minority class or the unique properties of the sample generation process. Through a systematic evaluation of all three areas, researchers can find out if synthetic data provides significant improvement in identifying rare cloud-based intrusions and determine which conditions under which it would provide little or negative results.

B. Operational Efficiency, Reliability, and Deployment Considerations

When considering the use of synthetic training data for cloud intrusion detection systems (IDS), there are several factors to consider. First, the operational efficiency of the system must be taken into account; second, how reliable the model will be over time as the characteristics of cloud-based workloads evolve; and third, whether or not synthetic augmentation can help improve the representation of less frequent types of attacks. It is true that synthetic training data generation can result in an increase in the number of samples used to train the IDS and allow for the development of more complex models. However, this process can lead to increased computational requirements. This can be problematic in cloud-based systems where large amounts of network traffic are being processed and/or the system has severe constraints on latency. Thus, when evaluating the effectiveness of using synthetic training data, researchers need to assess not only the classification performance of the IDS, but also its performance in terms of training time, memory usage, inference latency, and resource utilization. In general, it is better to have an IDS that provides a modest increase in recall (the percentage of true positives detected) at the expense of a significant decrease in processing requirement than one that achieves similar levels of detection efficacy with minimal additional processing overhead. Reliable performance is also critical because the characteristics of cloud-based traffic will continue to change as applications are updated, users begin accessing new services, and attackers develop new strategies. If synthetic examples generated from outdated or non-representative data sets are utilized, then the resulting model may not provide accurate representations of newly developed threats, which limits the model's capacity to generalize.

VI. CHALLENGES, LIMITATIONS, AND FUTURE DIRECTIONS OF SYNTHETIC DATA-BASED CLOUD INTRUSION DETECTION

The need for a Cloud Intrusion Detection System (CIDS) to consistently detect malicious activity in the context of the highly dynamic, complex environment of a cloud-based network. A potential solution to the issue of class imbalance is the use of synthetic training data. However, the success of this method depends on how well the synthetic data was produced, what type of samples were created, and how effective the classification algorithm being used to train on the data will be. A major concern is the lack of representative attack samples, especially in regard to those that are rare or emerging. If there is little to no information about an attack type in the original dataset, then synthetic production of data regarding that attack type will most likely result in a lack of realism and diversity in the generated samples. As such, the augmented dataset will probably only replicate biases that exist in the original dataset as opposed to creating new and useful insights. In addition to the fact that different cloud applications display different traffic patterns, authentication mechanisms and communications; therefore, prior

to using synthetic data generation techniques, researchers must examine the composition of the dataset, distribution of features, frequency of classes and types of attacks represented in the dataset.

There is another serious limitation regarding the accuracy and reliability of synthetic network traffic. Interpolation techniques like SMOTE can create new observations using interpolation, however, these generated new observations could represent combinations of features that would never exist in a real network environment. Generative models can represent complex statistical relationships better than interpolation, but can still create many repeated samples, miss rare attack signatures and/or create records that do not resemble real malicious traffic. All of these limitations can affect how well machine learning classifiers develop decision boundaries, and ultimately affect how effective a classifier will be at detecting previously unobserved events. As such, the quality of synthetic data should be assessed based on statistical comparison, analysis of individual features, measurement of diversity among generated samples, and domain-specific verification of sample validity. It is also essential to evaluate whether the generated samples retain any meaningful distinction between normal (benign) activity and malicious activity. Additionally, the size of a training dataset should not always equate to quality since more observations are valuable only if they help the model learn more effectively without introducing excessive noise or artifactual patterns.

The reliability of experiments is also a major concern for evaluating the contributions made by synthetic training data. When testing samples influence the way preprocessing occurs, how features are selected, how synthetic data is generated, or even the way that models are tuned; then there may be what is known as "data leakage." Data leakage can result in artificially high performance estimates and reduce the credibility of research findings. Therefore, in order to avoid this problem, datasets should be partitioned prior to performing synthetic augmentation, and data generation should only take place within the training portion of the dataset. Models should be selected based on performance on a separate validation partition of the dataset. Finally, the evaluation of the selected model should utilize only the untouched testing partition of the dataset. Performance evaluations should include minority class precision and recall, macro-F1 score, area under the precision-recall curve, and false positive rates. Experiments should be performed multiple times using different random seeds to determine if any improvements observed remain stable throughout each replication of the experiment. If appropriate, statistical significance tests can aid in determining if the observed differences in performance are real or just due to chance.

Future studies will evaluate adaptive methods of generating synthetic data that are able to react to new forms of attack and changing workloads of cloud-based applications. Generative models can be used in conjunction with domain specific rules to generate data samples which are more representative of real-world networks. Researchers may also investigate hybrid methods that combine synthetic data generation with cost sensitive learning, ensemble classification, feature selection and anomaly detection. These combinations can enhance the ability of a system to identify rare attacks while keeping false positive rates low and minimizing computational resources. Additionally, research should investigate applications running on containers, distributed cloud architectures, edge computing environments and multi tenant infrastructure; each has different traffic patterns and security needs. Privacy-preserving data generation techniques should be investigated when the dataset includes private details regarding users, organizations or network activities. However, the evaluation of privacy, sample utility, and computational resources must be done simultaneously as opposed to evaluating them individually. Finally, cross-dataset testing and evaluations over time can demonstrate if the model trained using synthetic data can generalize outside the scope of its training environment.

In summary, synthetic training data is a viable research avenue for enhancing the performance of cloud intrusion detection when classes are imbalanced; however, the effectiveness of this method must be demonstrated empirically and cannot simply be assumed. An optimal strategy includes realistic data generation, stringent experimental design, relevant evaluation measures, and explicit consideration of operational constraints. Researchers need to focus on detecting less-represented attacks without compromising precision, minimizing false alarms, and ensuring that the computational cost of model inference remains reasonable. Future empirical studies should help determine which augmentation techniques consistently enhance performance, which types of attacks continue to present problems for detection, and when synthetic data provides an advantage over traditional techniques. Reproducibility, transparency, and realism of testing environments are necessary to support the validity of the aforementioned conclusions. If researchers systematically address the above-mentioned issues, future cloud security systems may develop more reliable minority-class detection capabilities and therefore be able to better respond to the growing sophistication of cyber threats.

VII. EXPERIMENTAL FRAMEWORK AND PERFORMANCE ASSESSMENT

A. Dataset Preparation and Experimental Design

An experimental design for evaluating synthetic training data starts by choosing an appropriate intrusion detection dataset that contains both benign network traffic and various types of malicious activities. The chosen dataset should reflect typical network behavior such as attempted unauthorized access to the network; denial of service (DoS) attacks; scanning activities; and other patterns of intrusion that are commonly found in cloud-based environments. Preprocessing of the data will include identification of missing value(s); removal of inappropriate duplicate records; correction of inconsistencies in labelling; transformation of categorical variables into numerical representations compatible with the selected machine learning algorithms. If necessary, numerical variables will be scaled; and feature selection can help eliminate redundancies and computational complexity. Once the data has been preprocessed, it will then be divided into three parts: training, validation, and testing partitions. This division must take place prior to generating synthetic data to avoid potential information leakage from the training process to the final evaluation. The training partition will be utilized to develop the original baseline and augmented datasets, whereas the validation data will support model selection and parameter optimization. No modifications will be made to the testing partition prior to the final assessment. To determine how performance changes as malicious observations become increasingly scarce, several imbalance ratios should be investigated. All experimental conditions (i.e., data partitions, classification algorithms, and evaluation procedures) should remain identical across each balancing strategy. Appropriate baseline classifiers may include random forests, decision trees, and gradient boosting algorithms. Examples of synthetic generation techniques include random over-sampling, SMOTE, and generative models when applicable. Multiple random seeds should be considered to minimize the dependency on one experimental split. Overall, this experimental design creates a controlled environment for evaluating different training strategies and determining whether any improvement observed is due to synthetic data augmentation versus unrelated differences in experimental settings.

B. Classification Performance and Comparative Evaluation

Classification performance needs to be assessed based upon complementary performance measures reflecting the challenges in evaluating imbalanced intrusion detection datasets. Accuracy represents the percentage of correctly classified observations; however, it can mask weak recognition of rare attacks due to the dominance of benign traffic in the dataset. Precision represents the number of observations identified as malicious by the classifier that are truly malicious observations; recall represents the number of malicious observations correctly identified by the classifier. The F1 score is a combination of precision and recall; the macro-averaged F1 score considers each class equally irrespective of its frequency. Area under the Precision-Recall Curve is especially helpful in evaluating performance on minority classes; False Positive Rate represents how often benign traffic is incorrectly identified as malicious. Each classifier should be trained on the original imbalanced dataset and the various alternative augmented datasets under similar conditions. Results should be recorded individually for each type of attack as an augmentation technique may enhance detection of some types of attacks but not others. Performance variations should be assessed across multiple levels of imbalance and numerous times experimental trials were conducted. Statistical analysis should be utilized, when appropriate, to verify if improvements obtained are consistent and statistically significant. Measurements relating to computation (i.e., training time, inference latency, memory usage) should also be reported. A comparative evaluation should determine whether synthetic augmentation results in balanced improvements in detection quality and operational efficiency. The evaluation process described above does not rely on a single performance measure and provides a more comprehensive assessment of the practical worth of synthetic training data for cloud intrusion detection systems.

C. Interpretation of Results and Research Implications

When interpreting the results of experiments with synthetic training data, researchers need to consider not only how much the synthetic training data improved the performance of the classification algorithm, but also why the improvement occurred, and how consistent that improvement will be across various conditions. For example, if there was an increase in the recall of the minority class, it could mean that the augmented model is better at identifying rarely occurring attack signatures. On the other hand, if there was a large decrease in precision or an increase in false positives, then the overall security improvement would be minimal. Also, an increase in the macro F1-score should be analyzed in conjunction with individual class performance measures to determine if certain attack types are still being missed. Researchers should analyze generated samples for diversity, statistical consistency, and the relationship between network characteristics. Researchers should compare their results to random oversampling and cost-sensitive learning to demonstrate that generating synthetic samples provides benefits beyond just increasing the weight of the minority class.

Finally, researchers should test their model on independent testing data, and if possible, use other datasets or different time frames representing changes in the network environment to assess generalization capabilities. Researchers should also consider the computational costs associated with using complex generative models, since the additional resource requirements may not provide significant improvements in detection capability. Researchers should clearly document all experimental measurements and differentiate between those results that were empirically verified and those that were expected based upon theoretical assumptions. If researchers find that generating synthetic samples results in inconsistent performance, then those results are still valuable as they provide insight into why the generation method, dataset, or classification algorithm did not perform as well as expected. In summary, synthetic training data should be chosen based upon empirical evidence rather than solely on novelty. Through a combination of quantitative performance evaluation, sample quality evaluation, and deployment considerations, the proposed framework can enable decision-makers to make informed choices as to when synthetic training data is appropriate for improving intrusion detection in cloud computing environments.

VIII. ADVANCED CONSIDERATIONS FOR SYNTHETIC DATA-DRIVEN CLOUD INTRUSION DETECTION

A. Feature Representation and Attack Pattern Learning

Effective cloud intrusion detection requires machine learning models to be able to detect significant differences in legitimate network traffic and malicious behavior. There are many types of features in network traffic datasets; for example, connection length, number of packets transmitted, rate at which bytes were transmitted, protocol type, service information, outcome of authentication attempts, and evidence of anomalous communications. The quality of each of the features determines how well synthetic data generations and classifications can be done. When critical aspects of an attack are not well-represented, creating more samples will likely have minimal effect on improving detection performance. Feature engineering can assist in finding the best features, eliminating unneeded variables, and creating better representations of attacks that occur less frequently. It is important to perform feature selection on the training data only so that there is no "information leakage" from the testing data into the training process. Different machine learning algorithms may react differently to different types of scaling of numbers, encodings of categorical variables, and correlations among variables. As such, all decisions made about preprocessing should be recorded and applied in the same manner regardless of the experimental condition.

Synthetic data generation is better off preserving the relationship among the various characteristics of a network (e.g., packet counts, connection duration, and transmission rates) than it is reproducing each characteristic separately. As such, if we were to generate synthetic data based on the characteristics individually, we could end up with some very odd-looking data that would ultimately undermine the reliability of the classifier. Therefore, researchers need to evaluate how well the relationships among the characteristics are preserved by evaluating the correlation among the characteristics, how well each characteristic represents its respective class, and how diverse the generated samples are. This will allow researchers to assess whether the newly generated records are truly representative. In addition to using statistical methods to validate the newly generated records, researchers can also utilize domain knowledge to help identify those characteristics that appear to be unrelated to the problem being studied. This approach allows researchers to learn models that are more reliable and interpret their results more easily.

B. Robustness Against Evolving and Previously Unseen Attacks

As organization continues to evolve and develop new applications; alter its network configuration; add more services; and react to evolving security threats, the cloud environment will continue to change. Such changes may lead to "distribution shifts" which could result in a drop in the performance of previously trained intrusion detection models. While synthetic training data may help to improve recognition of known minority attack types, it is unlikely to automatically ensure that the model will effectively identify attacks that were not present in the original training data. Generative approaches to training models typically use information from existing data to train the model, so their ability to accurately represent unfamiliar threats is limited by the amount of information used during training. Therefore, when evaluating the effectiveness of a model that was trained with augmented data, you should evaluate the model under conditions that represent changes in the characteristics of traffic and the types of attacks that occur. Testing over time; evaluating across multiple datasets; and testing the model against attack categories that were held out of the training process are all methods that can support evaluating whether a model is robust enough to adapt to changes in the environment.

An evaluation of synthetic data should also look at whether creating overly precise decision boundaries using synthetic data reduces adaptability to changes in traffic patterns. While models trained on randomly partitioned data often perform well, they are frequently less successful when evaluated against traffic collected at a different time. Evaluations conducted in multiple environments can highlight these issues. Researchers may evaluate combining supervised classification with anomaly detection

to create models capable of identifying anomalies that do not fall into previously defined attack types. Monitoring of models periodically may indicate decline in recall, increase in false positives, or change in distribution of features in the input space. Retraining of models under controlled conditions may allow for incorporating representative new data points. These types of evaluations will ensure that synthetic data is used as part of a comprehensive security strategy and not as a replacement for continuous threat monitoring.

C. Deployment Optimization and Security Decision-Making

Deploying synthetic data-enhanced intrusion detection models will require finding a balance between classification accuracy and processing speed, resource usage, scalability and alert management. In order to effectively deploy these models on cloud infrastructure that handles millions of network events, the latency of inference and the computational efficiency must be considered. While advanced generative models can enhance the diversity of training data, the computational cost associated with them can be high; especially if the minority attacks are very rare or the feature space is large. Therefore, a practical approach to deployment should evaluate whether the advantages of synthetic data augmentation are worth the added time and resources needed for generating the synthetic data, training the model, and maintaining it. Simple classification algorithms may be used to achieve acceptable performance for applications that have tight latency requirements. However, if the benefit of using more complex models outweighs the extra overhead, then they may be more suitable.

False positive and false negative consequences should be factored into security decision making. A high number of false positive alerts could lead to analyst burnout; conversely, a low number of false negative alerts could lead to undetected malicious activities. Rather than relying on a predetermined classification threshold, an organization's security goals and workload should determine its desired operating threshold. Minority class recall, precision, alert volumes, inference latencies, and changes in traffic patterns should all be monitored as part of performance monitoring. Additionally, deployment processes should protect sensitive network information and prevent generated samples from revealing sensitive aspects of the original dataset. Organizations can increase trust in the resulting system through transparency in documentation, reproducibility in experiments, and regular external evaluations. Synthetic training data will be most effective when used as part of a thoroughly evaluated detection process that incorporates realistic data preparation, strong classification, efficient deployment, and continuous security oversight.

IX. COMPARATIVE EVALUATION OF SYNTHETIC DATA TECHNIQUES FOR CLOUD INTRUSION DETECTION

Synthetic training data is a useful way to deal with the problem of class imbalance in cloud intrusion detection systems. If malicious traffic only accounts for a very small amount of all network observations, then machine learning models will naturally focus on identifying legitimate traffic and miss rare attacks. Generating synthetic data will help to increase the number of examples from minority-class attacks and could lead to better classification performance. However, there are many factors that determine which method will work best including the type of data used to train the model, the properties of the minority-class samples, and the type of learning algorithm being applied.

Table 2: Comparison of Training Techniques for Imbalanced Data Classification

Training technique	**Data generation approach**	**Main advantage**	**Major limitation**	**Evaluation metric**
Original dataset	Retains natural class distribution	Establishes baseline performance	Minority attacks may be overlooked	Macro-F1
Random oversampling	Duplicates minority-class records	Simple implementation	Risk of overfitting	Recall
SMOTE	Interpolates between minority samples	Expands minority-class representation	May produce unrealistic combinations	Precision and recall
Generative models	Learns patterns to create synthetic records	Can capture complex relationships	Training instability and sample-quality concerns	Precision-recall AUC
Cost-sensitive learning	Assigns different penalties to classification errors	Avoids synthetic sample generation	Requires appropriate cost adjustment	F1-score and false-positive rate

The use of a comparative approach demonstrates how different approaches to dealing with class imbalances have varying effects on the ability to identify classes. A base-line of comparison is required in order to determine if balancing the data will improve classification performance. Random oversampling increases the amount of exposure the minority-class has to the algorithm, however it does so without generating any additional data. On the other hand, SMOTE creates synthetic samples based upon interpolation from existing data points which can provide the classifier with the ability to define larger decision boundaries. Generative models allow for representation of complex relationships among network features, however validation is essential prior to usage. Cost-sensitive learning is an alternative mechanism to modify error costs as opposed to altering the training data set. As such, the best approach should be determined via experimental comparison as opposed to assuming superiority.To ensure consistency in evaluation of each method, evaluations must use the same training, validation, and testing partitions. All synthetic data must come from only the training partition. No synthetic data may be generated from the testing partition prior to the final evaluation. The evaluation metrics must include the minority class precision, recall, macro-F1, area under the precision-recall curve, and false-positive rate. In addition, the training time, inference latency, and memory usage during evaluation should be measured. Experiments must be repeated to determine if the improvement shown by the selected method is consistent regardless of the level of imbalance or random seed used.

X. PRACTICAL IMPLICATIONS AND FUTURE RESEARCH DIRECTIONS

A. Data Quality and Synthetic Sample Validation

Synthetic training data can have a direct impact on how reliable a cloud-based intrusion detection system will be. It is important to generate training data that includes the same types of relationships between various aspects of the network (packet statistics, connection duration, protocol characteristics, etc.) as the actual data would. Synthetic training data with too many unrealistic combinations of features can lead classifiers to pick up on misleading patterns and produce unreliable results. Thus, researchers need to evaluate the training data's statistical distribution, feature correlation, sample diversity and class label consistency prior to utilizing it in their training set. Researchers must also check for duplicates in the training data and excessive similarity to previously generated samples. Because interpolation-based methods and generative models create different kinds of errors when generating training data, there may be multiple ways to evaluate the quality of the generated training data. Lastly, the quality of the generated training data should not be determined solely based upon its similarity to the original dataset. Rather, it should be evaluated in terms of its ability to be used in independent intrusion detection experiments which measure both the ability to recognize minority classes and the ability to generalize.

B. Improving Minority-Class Detection Performance

Synthetic data generation offers a viable option for addressing the issue of learning from imbalanced data; however, it must be used cautiously to avoid distorting the underlying data distribution and increasing false positive rates. When using synthetic data generation, researchers must consider the appropriate level of augmentation (moderate vs. extensive) based upon the number of samples available for each minority class. Additionally, researchers should compare the results of using synthetic data generation to other methods such as random oversampling and cost-sensitive learning. Lastly, researchers should evaluate the impact of synthetic data generation on each of the minority classes as well as overall classification performance. Ultimately, researchers need to identify which classification threshold will provide an acceptable balance between detecting rare attacks and minimizing the volume of alerts generated. In general, researchers should select strategies that yield consistent results across numerous experiments rather than relying on a single experiment to produce optimal results.

C. Computational Efficiency and Cloud Deployment

Computational efficiency is critical to consider while deploying intrusion detection models in cloud environments. The ability to process large amounts of traffic in a short amount of time will allow for rapid prediction of threats; however, if the model consumes too much memory, it will not be effective. The use of advanced generative models that generate many different types of synthetic observations can increase development costs and delay updates to the model due to the long training times. In contrast, simpler methods such as random oversampling and SMOTE may be better options when they result in similar levels of threat detection. When conducting research on various models, researchers should compare training times, inference latencies, resource consumption, and performance across all models under similar conditions. Additionally, researchers should also consider the impact of false alarms. Too many false alarms can put a strain on security analysts and ultimately lead to decreased efficiency in responding to incidents. lead to significant improvements in detection capabilities.

D. Future Research and Adaptive Detection Systems

In the future, researchers should look into adaptive synthetic data generation methods that are based on how the cloud traffic changes and the new attacks. It is possible that the models trained with the previous historical data will lose effectiveness as applications, network configuration and attack behavior evolve. To see if the classifiers enhanced with the synthetic data can be generalized beyond the environment where they were developed, researchers should use time-based evaluation and testing across different datasets. Hybrid systems that combine synthetic data generation with anomaly detection, cost sensitive learning and ensemble classification could be explored by researchers. These types of systems could provide improved identification of rare attacks and support detection of unknown behaviors. Researchers should investigate other privacy preserving generation methods when the network datasets include sensitive information. To ensure that future studies continue to produce dependable cloud intrusion detection systems for developing cyber security requirements, researchers need to focus on producing repeatable experiments, conducting independent testing and transparently reporting results.

XI. CONCLUSION

The rapid growth in the adoption of cloud computing has driven the need for more reliable intrusion detection systems (IDS) that can accurately detect the increasing complexity of cyber threats occurring within both distributed and dynamically changing environments. While machine learning based intrusion detection provides a promising approach to detecting malicious activity, there are many challenges associated with the effectiveness of these approaches due to the inherent problem of class imbalance in most network traffic data sets. Specifically, when a large majority of connection attempts made during the time period used to collect the network traffic are legitimate, classification models will likely make decisions that favor the majority class over the minority class; i.e., the model will not identify rare but damaging attacks. To address the challenge of class imbalance in cloud computing intrusion detection, this research titled Measuring the Impact of Synthetic Training Data on Cloud Intrusion Detection Under Class Imbalance investigated the use of synthetic data generation as a means to alleviate the effects of class imbalance. The study developed a methodological framework to evaluate how various approaches to balancing the distribution of classes affect classification performance, minority class identification, computational overhead, and generalizability of the model. Furthermore, the study demonstrated that improving the ability to represent rare attacks through synthetic data generation is more complex than simply adding more training examples to the model. Rather, the generated examples must contain meaningful relationships among the various attributes of the network, and the learned patterns must allow the model to differentiate malicious activities from those that represent normal network behavior. Therefore, the development and evaluation of synthetic training data should be viewed as part of a broader strategy for enhancing intrusion detection capabilities in cloud computing rather than as a stand-alone guarantee for improving the overall security posture of the organization.

The proposed research framework will evaluate various methods such as; (1) Training with naturally imbalanced data, (2) Random oversampling of the minority class, (3) Synthetic Minority Over-sampling Technique (SMOTE), and (4) Generative methods as necessary. Additionally, the research framework will recognize Cost Sensitive Learning as a method for comparison. Cost Sensitive Learning alters the impact of misclassification errors, but does not create new sample points. To assess intrusion detection effectiveness in a balanced manner, the evaluation framework will use Precision, Recall, F1-Score, Macro-Averaged Performance Measures, Area Under the Precision-Recall Curve, and False Positive Rates. Emphasis will be placed upon Minority Class Recall, as missing rare attack types could lead to significant security vulnerabilities in cloud computing environments. In order to determine if a specific method can be deployed cost-effectively, computational metrics including Training Duration, Memory Requirements, and Inference Latency will be evaluated. Finally, the research methodology includes several safeguards to prevent data leakage and inaccurate performance estimations. These safeguards include appropriate data partitioning, synthetic generation of training data only, independent testing, and consistent experimental conditions. Because experimental results depend on the dataset(s) used, the algorithms applied, and the experimental setup, the ultimate determination of the effectiveness of the various methods must be based on empirical testing rather than on prior assumptions.

A major finding of this study is that generating synthetic data offers both significant advantages and major limitations. For example, techniques such as SMOTE can help to improve representation of the minority class, whereas generative models can potentially find more intricate relationships between the various elements of network traffic. However, synthetic samples can generate unnatural combinations of elements, continue to reflect current biases, and/or be unable to accurately represent the variety of types of legitimate attacks. Furthermore, excessive use of data augmentation can result in higher rates of false positives, and can also lead classifiers to identify patterns that are purely artifacts of the augmentation process, and therefore do not apply to previously unobserved network traffic. Therefore, researchers must evaluate sample quality, class specific performance, and resilience across various levels of class imbalance prior to advocating for a specific approach. Additionally,

evaluating model performance against other independent datasets, or data collected from different time periods can provide further evidence regarding the ability of the model to generalize. Given that cloud environments are continually evolving due to the development of new applications, changes in workloads, and evolving attack methods, continuous performance evaluation will be required. Finally, when making decisions concerning how to deploy a system, there must be consideration given to balancing the ability of the system to detect threats with the cost of computing, the number of generated alerts, and response requirements. Synthetic augmentation combined with careful selection of features, appropriate model selection, and continual validation may offer a more robust solution than reliance on any single balancing approach.

XII. REFERENCES

[1] Alazab, M., Abawajy, J. and Islam, R. (2012) 'Improving the detection of malicious web requests using cluster-based intrusion detection', Journal of Network and Computer Applications, 35(6), pp. 1958–1966.

[2] Amachaghi, E., Abdulkareem, S., Foh, C. and Shojafar, M. (2025) 'Improving intrusion detection in O-RAN with synthetic data generation: A GAN and SMOTE approach', Telematics and Informatics Reports, 20, article 100269.

[3] Anderson, J.P. (1980) Computer Security Threat Monitoring and Surveillance. Technical report. Fort Washington, PA: James P. Anderson Company.

[4] Breiman, L. (2001) 'Random forests', Machine Learning, 45, pp. 5–32.

[5] Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P. (2002) 'SMOTE: Synthetic minority over-sampling technique', Journal of Artificial Intelligence Research, 16, pp. 321–357.

[6] Chen, T. and Guestrin, C. (2016) 'XGBoost: A scalable tree boosting system', in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, pp. 785–794.

[7] Choi, H., Kim, M., Lee, G. and Kim, W. (2019) 'Unsupervised learning approach for network intrusion detection system using autoencoders', Journal of Supercomputing, 75, pp. 1635–1660.

[8] Dal Pozzolo, A., Caelen, O., Le Borgne, Y.-A., Waterschoot, S. and Bontempi, G. (2015) 'Learned lessons in credit card fraud detection from a practitioner perspective', Expert Systems with Applications, 41(10), pp. 4915–4928.

[9] Dina, A.S., Siddique, A.B. and Manivannan, D. (2022) 'Effect of balancing data using synthetic data on the performance of machine learning classifiers for intrusion detection in computer networks', arXiv preprint, arXiv:2204.00144.

[10] Ferrag, M.A., Maglaras, L., Moschoyiannis, S. and Janicke, H. (2020) 'Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study', Journal of Information Security and Applications, 50, article 102419.

[11] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. (2014) 'Generative adversarial nets', in Advances in Neural Information Processing Systems 27. Red Hook, NY: Curran Associates, pp. 2672–2680.

[12] He, H. and Garcia, E.A. (2009) 'Learning from imbalanced data', IEEE Transactions on Knowledge and Data Engineering, 21(9), pp. 1263–1284.

[13] Japkowicz, N. and Stephen, S. (2002) 'The class imbalance problem: A systematic study', Intelligent Data Analysis, 6(5), pp. 429–449.

[14] Khraisat, A., Gondal, I., Vamplew, P. and Kamruzzaman, J. (2019) 'Survey of intrusion detection systems: Techniques, datasets and challenges', Cybersecurity, 2, article 20.

[15] Krawczyk, B. (2016) 'Learning from imbalanced data: Open challenges and future directions', Progress in Artificial Intelligence, 5, pp. 221–232.

[16] Lin, W.-C., Tsai, C.-F., Hu, Y.-H. and Jhang, J.-S. (2017) 'Clustering-based undersampling in class-imbalanced data', Information Sciences, 409–410, pp. 17–26.

[17] Mirsky, Y., Doitshman, T., Elovici, Y. and Shabtai, A. (2018) 'Kitsune: An ensemble of autoencoders for online network intrusion detection', in Proceedings of the Network and Distributed System Security Symposium. Reston, VA: Internet Society.

[18] Moustafa, N. and Slay, J. (2015) 'UNSW-NB15: A comprehensive data set for network intrusion detection systems', in Proceedings of the Military Communications and Information Systems Conference. Canberra: IEEE.

[19] Niyaz, Q., Sun, W. and Javaid, A.Y. (2016) 'A deep learning based DDoS detection system in software-defined networking', EAI Endorsed Transactions on Security and Safety, 3(9).

[20] Paszke, A. et al. (2019) 'PyTorch: An imperative style, high-performance deep learning library', in Advances in Neural Information Processing Systems 32. Red Hook, NY: Curran Associates, pp. 8024–8035.

[21] Ring, M., Wunderlich, S., Grüdl, D., Landes, D. and Hotho, A. (2019) 'A survey of network-based intrusion detection data sets', Computers & Security, 86, pp. 147–167.

[22] Saito, T. and Rehmsmeier, M. (2015) 'The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets', PLOS ONE, 10(3), article e0118432.

[23] Salamon, J. and Bello, J.P. (2017) 'Deep convolutional neural networks and data augmentation for environmental sound classification', IEEE Signal Processing Letters, 24(3), pp. 279–283.

[24] Sharafaldin, I., Lashkari, A.H. and Ghorbani, A.A. (2018) 'Toward generating a new intrusion detection dataset and intrusion traffic characterization', in Proceedings of the International Conference on Information Systems Security and Privacy. Setúbal: SciTePress, pp. 108–116.

[25] Tavallaee, M., Bagheri, E., Lu, W. and Ghorbani, A.A. (2009) 'A detailed analysis of the KDD CUP 99 data set', in Proceedings of the Second IEEE Symposium on Computational Intelligence for Security and Defense Applications. Ottawa: IEEE, pp. 1–6.

[26] Van der Maaten, L. and Hinton, G. (2008) 'Visualizing data using t-SNE', Journal of Machine Learning Research, 9, pp. 2579–2605.

[27] Van Vlasselaer, V., Bravo, C., Caelen, O., Eliassi-Rad, T., Akoglu, L., Snoeck, M. and Baesens, B. (2015) 'APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions', Decision Support Systems, 75, pp. 38–48.

[28] Wang, S., Liu, T. and Wang, L. (2020) 'An effective convolutional neural network based on SMOTE and Gaussian mixture model for intrusion detection in imbalanced dataset', Computer Networks, 177, article 107315.

[29] Xu, L., Skoularidou, M., Cuesta-Infante, A. and Veeramachaneni, K. (2019) 'Modeling tabular data using conditional GAN', in Advances in Neural Information Processing Systems 32. Red Hook, NY: Curran Associates, pp. 7335–7345.

[30] Zhang, Y., Chen, X., Jin, L., Wang, X. and Guo, D. (2019) 'Network intrusion detection: Based on deep hierarchical network and original flow data', IEEE Access, 7, pp. 37004–37016.