Towards Developing Generative Adversarial Networks based Robust Intrusion Detection Systems for Imbalanced Dataset using Hadoop-PySpark

Md Abdur Rahman[1*] and Hossain Shahriar[2]

[1]Department of Intelligent Systems and Robotics, University of West Florida
11000 University Parkway, Pensacola, 32514, FL, United States.
[2]UWF Center for Cybersecurity, University of West Florida
220 W Garden St Suite 250, Pensacola, 32502, FL, United States
mr252@students.uwf.edu, **hshahriar@uwf.edu**
.

Abstract

The rise of machine learning has inspired the development of machine learning-based intrusion detection systems (IDS), yet their performance diminishes particularly with imbalanced data and missing datasamples. Generative Adversarial Networks (GANs) offer a solution by handling imbalanced data and addressing new vulnerabilities and challenges in cybersecurity and privacy for anomaly detection. GANs generate synthetic data to augment limited samples, enhancing the training of machine learning models to detect novel anomalies and patterns. This paper proposes an efficient GAN-based Random Forest (RF) model to introduce novel defense techniques with distributed computing using PySpark for big data processing because of computations efficiently in cluster of machines. Experimental results demonstrate that integrating synthetic data with existing datasets enhances model stability and robustness in detecting cyber attacks, with an accuracy of detection rate 99.94% when employing GANs generated augmented training dataset.

Keywords: Intrusion detection, Imbalaced datasets, Random forest, Generative adversarial networks, Pyspark

1 Introduction

Cybersecurity has risen as a significant concern in computing systems. While there are extensive methods to boost network protection [1], current approaches fall in problems in ensuring security against evolving threats. For instance, in May 2017, ransomware targeted numerous devices across 150 nations, affecting over 300,000 individuals [2].

Typically, network intrusions handled by known attacks involve misuse behaviours and anomalous activities for adversarial attacks. Misuse behaviours can be identified using malevolent traffic patterns, while anomaly detection recognizes unknown attacks by using historical data [3]. Lately, there was a lot of research on hybrid detection, combining anomaly-based and signature-based methods to improve attack detection and reduce false alarms [4-6].

Therefore, it is essential to identify and detect known attacks as well as unknown anomalies efficiently, simultaneously. Based on the above issues, the goal of IDS is to detect whether it involves malicious activity. Most prior works used conventional machine learning algorithms for intrusion detection: Support Vector Machine [7-12], Decision Tree [13], Naive Bayes [14], and Random Forest [15]. Also, Akter et al., 2023 used difefernt machine learning approaches for the prediction of risk factor for elements of cryptocurrency market [17]. Moreover, quantum generative adversarial networks is used in intrusion detection system to detect adversarial attacks at TCP/IP for detecting anomalies [18].

Because of remarkable power of deep learning, several research works used neural networks in order to detect it by Recurrent Neural Network [19], Multi-layer Perceptron [20], and Convolutional Neural Network [21]. Further, one work selects features and used hybrid algorithm for improved accuracy IDS [22]. Despite the significant advancements made by prior methods, the presence of imbalanced datasets remains a new challenge which hinders performance of IDS [23]. This problem happens in the case of lower number of datasamples regarding intrusions compared to normal. This issue encounters difficulties in intrusion detection since routine activities are widespread in authentic networks. Moreover, machine learning and deep learning (ML/DL) classifiers became in danger to adversarial attacks generated samples for fooling discriminators in classification reports. Since IDSs were created with ML/DL approaches for classification, there capacity becomes questionable.

Because of its capability to generate similar samples, GAN is used in various domains [24]: computer vision [25], dynamic networks [26] natural language processing [27], and security [28]. The class imbalanced issue is effectively addressed by the high-dimensional capabilities of GANs. For instance, a conditional GAN [29] was used by Douzas which results in generating minority datasamples [30]. One research used AC-GAN for the classification of network traffics which effectively addresses the imbalanced problem [31].

NSL-KDD is a imbalanced datasets and datasample for Normal traffic samples is majority class and attack categories is minor class. Especially User to Root (U2R) and Remote to Local (R2L) are minority classes [32]. Moreover, Rahman et al., 2023 used K-means clustering to make the datasamples as clustered input to the Random Forest classifier for big data distributed systems for detecting in a high accuracy using big data processing because it was introduced to use computing capabilities across clusters of machines in the case of huge amount of data[16].

Therefore, we have focused to enhance this work to address the imbalanced dataset problem using GANs by generating datasamples for specific classes so that the imbalanced dataset issues can be resolved for IDS in distributed computing using PySpark. Moreover, the new types of instances is generated which might be either normal or any of existing attacks. When the model is trained with the combination of existing and synthetic dataset based on GAN, it could be well trained and able to detect the attacks used new type of instances which are not like the historical data.

The remaining paper is organized as follows: the dataset was described in Section 2. ML and Scope of Big Data on network security is included in Section 3. Proposed Model is well explained in Section 4. Results and discussion was described in Section 5. Finally, Section 6 concludes with the results as a summary of this.

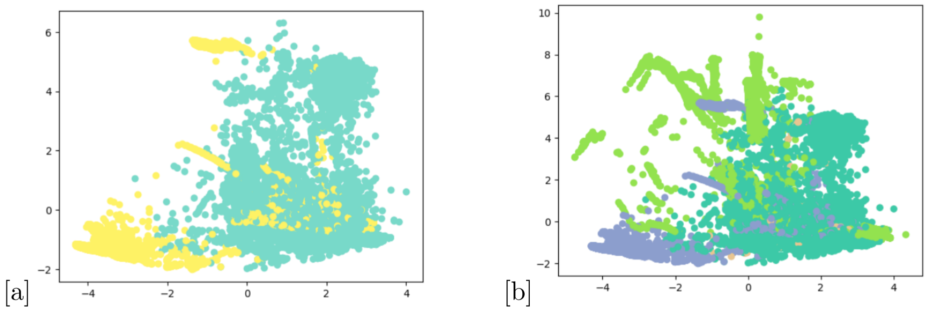

Fig. 1 Generated clusters after applying PCA for: (a) Normal and DoS (b) Normal-DoS-Probe-U2R-R2L.

2 Dataset

The NSL-KDD dataset, an enhanced version of the KDD Cup 1999 dataset, serves as a pivotal resource for evaluating intrusion detection systems (IDS). It comprises a diverse array of features extracted from network traffic data, including protocol type, service, and flags, among others. These features provide valuable insights into network behaviour for using to detect anomalous activities.

In terms of scale, NSL-KDD dataset have 125,973 records for training and 22,544 records for testing. Despite its huge size, the dataset is characterized by class imbalanced as the disparity is observed between the number of normal instances and intrusion instances. It is observed that datasample for Normal traffic samples is majority class and attack categories is minor class. Especially, Remote to Local and User to Root are minority classes. In this work, we will resolve this imbalanced problem and make it with distributed capabilities for making accuracy better. Fig. 1 shows on how clusters were generated after applying principle component analysis between Normal and DoS (a) as well as Normal and other 4 attacks (b).

This imbalance happens due to the nature of network traffic, where real normal activities significantly outnumber malicious intrusion attempts. Consequently, the dataset contains a lower number of intrusion instances compared to normal instances. This class imbalance poses a challenge for intrusion detection algorithms, as they may exhibit a bias towards the majority class (normal instances) during training, potentially leading to decreased performance in detecting and accurately classifying intrusion instances.

Addressing this class imbalance is important for developing robust intrusion detection models that can effectively identify and classify both normal and intrusive activities. Researchers often employ various techniques, such as sampling methods or algorithmic adjustments, to mitigate the effects of class imbalance and enhance the performance of IDS models trained on the NSL-KDD dataset. Despite its imbalance, the NSL-KDD dataset remains good dataset if we can generate intrusion instances for training and evaluating IDS models in making cybersecurity research advanced.

3 Machine Learning Methods & Big Data Tools

3.1 K-means

Clustering analysis has gained prominence as a favored technique for unsupervised data grouping, offering advantages such as eliminating the need for artificial divisions and external supervision. It aims to uncover inherent structures within the data by grouping similar data points together based on their proximity and maximizing similarity within each cluster. One

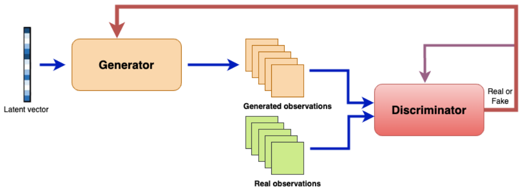

Fig. 2 The architecture of a typical GANs model with generator (G) and discriminator (D)

widely used algorithm for clustering is K-means, a popular unsupervised ML approach [33]. K-means clustering is a method that organizes the master dataset into K clusters. During this process, each observation is allocated to the cluster with the closest mean value. This grouping strategy ensures that data points with similar traits are placed together, making it possible to uncover separate clusters within the dataset.

3.2 Random Forest

The Random Forest algorithm plays a pivotal role in cybersecurity detection, offering robust capabilities for identifying and mitigating potential threats. In this context, Random Forest leverages its ability to construct multiple decision trees, each trained on different data samples, enhancing the model's resilience against diverse cyber threats. By amalgamating the outputs from these trees, the algorithm provides a comprehensive and accurate classification of network traffic or system behavior. Its adaptability to handle both classification and regression tasks makes Random Forest a formidable tool in cybersecurity, contributing to the creation of effective Intrusion Detection Systems (IDS) and bolstering defenses against evolving cyber threats.

3.3 Generative Adversarial Networks

Generative Adversarial Networks (GANs) are a significant impacts in deep learning, enabling computers to create fresh data. GANs create new instances of data resembling their original training data by learning patterns within that data. Initially applied to existing datasets, GANs operate the framework, comprising two components: the generator and discriminator. Fig. 2 depicts a typical architecture of GANs model. In this adversarial attacks, the generator produces new instances to training discriminator. Simultaneously, the discriminator, utilizing inputs from the original dataset and the output of the generator, predicts the probability of whether the input is real or fake. This interplay between the generator and discriminator reaches a local Nash equilibrium, optimizing their respective costs. The generator minimizes its cost by creating adversarial data that the discriminator incorrectly classifies as real, while the discriminator minimizes its cost by accurately distinguishing between real and fake data. This adversarial process enables GANs to generate realistic data instances.

3.4 Handling Imbalanced Dataset (GANs)

Generative Adversarial Networks (GANs) consist of generator (G) and a discriminator (D). The generator creates artificial samples by introducing random noise z and utilizing them as inputs $S(z)$. The discriminator assesses the probability $D(x)$ that a given sample x originates from the genuine distribution. Initially, the Jensen–Shannon (JS) divergence was proposed to gauge this similarity:

$$J_S(p_{data}|p_g) = \frac{1}{2}KL(p_{data}|p_m) + \frac{1}{2}KL(p_g|p_m) \tag{1}$$

The Jensen–Shannon (JS) divergence is initially introduced to measure similarity which is used to calculate using the Kullback-Leibler (KL) divergence between the true data distribution (pdata) and a generated distribution (pg). This measure is used to assess the similarity between two probability distributions. The maximized data is expressed as follows:

$$\min_G \max_D \mathbb{V}(D, G) = \min_G \max_D \left(\mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]\right) \tag{2}$$

In the context of generative adversarial networks (GANs), the maximized data is represented as the minimization of the divergence between the discriminator (D) and the generator (G). This formulation aims to minimize the difference between the distribution of real data and the distribution generated by the generator.

3.5 Pyspark

Pyspark, a widely acclaimed big data processing framework, has garnered significant attention for its efficient handling of large-scale datasets. As an open-source framework built upon Apache Spark, it has become a cornerstone in the IT industry for robust big data processing.

Pyspark, an open-source framework rooted in Apache Spark, stands out as a prominent choice for large-scale data processing in the IT sector. Leveraging the capabilities of Apache Spark, one of the most widely adopted big data processing frameworks, Pyspark empowers developers to efficiently process extensive datasets in parallel, establishing itself as a pivotal tool in the realm of big data processing.

PySpark is introduced to facilitate the integration of Apache Spark and Python, serving as a Python API for Spark. It enables interaction with Resilient Distributed Datasets (RDDs) within Apache Spark using Python. This seamless interaction is made possible through the utilization of the Py4j library. The Py4J library, seamlessly integrated into PySpark, enables dynamic interfacing between Python and JVM objects. PySpark provides several libraries for writing efficient programs, and it supports compatibility with various external libraries. Some notable examples include: PySparkSQL, MLlib, GraphFrames.

The Spark API (Application Programming Interface) facilitates distributed computation and it supports officially 4 languages: Java, Scala, Python, and R. It offers various operators such as map and reduce for efficient data processing. At its core, Spark comprises Spark Core, providing fundamental functionality for defining Resilient Distributed Datasets (RDDs) and executing operations like map, filter, and reduce.

Built on this foundation, Spark's additional libraries include:

Spark SQL: Tailored for SQL and structured data processing, it represents each database table as an RDD, with Spark SQL queries seamlessly translated into Spark operations.

MLlib: This library houses commonly used machine learning algorithms, implemented as Spark operations on RDDs. While MLlib is transitioning to ML, a more advanced Spark Machine Learning toolset, the current library includes scalable algorithms for classifications, regressions, etc. ML provides an elevated API constructed on DataFrames, adopting the pipeline concept influenced by the scikit-learn project.

GraphX: Dedicated to graph manipulation and parallel graph operations, GraphX provides a collection of algorithms and tools for effective graph computations.

Spark Streaming: It is scalable and fault-tolerant stream processing for real-time data, and ensures high-throughput and adaptability in handling streaming data efficiently.

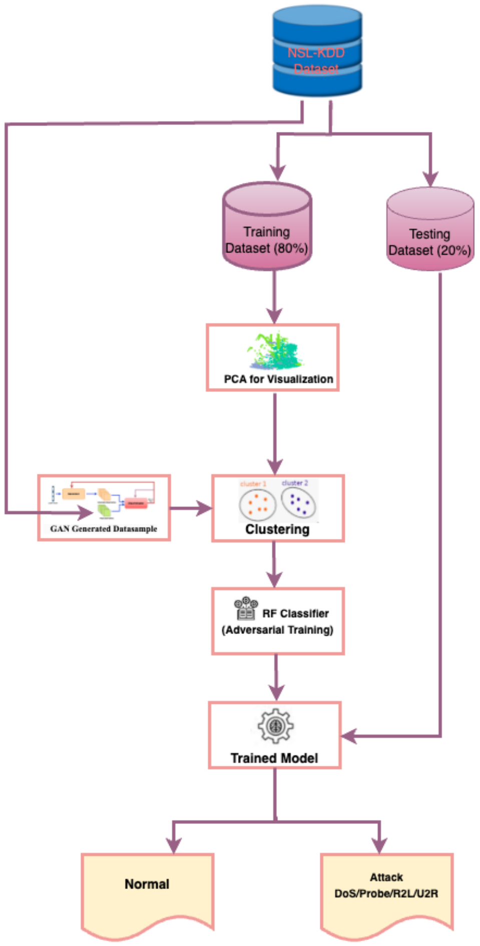

Fig. 3 GANs based Random Forest Model for training for comprehensive IDS

4 Proposed Model

The proposed method for Intrusion Detection System (IDS) is a multi-step approach to effectively handle imbalanced data and enhance classification accuracy. The flowchart in Fig. 3 illustrates how the proposed GANs based RF model augments artificial dataset with existing dataset for comprehensive IDS training and testing. Initially, a Random Forest (RF) classifier is trained using an existing dataset for focusing intrusion detection efficiently. Due to the inherent class imbalance within the dataset, particularly with fewer samples representing intrusions, a Generative Adversarial Network (GAN) is employed in the second step.

The GAN framework comprises generators and discriminators, with the aim of generating additional data samples specifically for the intrusion class. These synthetic/artificial samples are then integrated with the existing dataset for clustering purposes. The resulting clusters, representing both real and generated samples, are utilized for adversarial training of the RF classifier. This process ensures comprehensive coverage and robustness, minimizing the possibility of evasion by potential intrusions.

Once the RF classifier is trained with the combined dataset, comprising both real and generated samples, it undergoes evaluation using testing dataset. This evaluation step serves to assess the model's predictive capabilities, distinguishing between normal observations and potential attacks with high accuracy. Through this constructive approach integrating RF classification, GAN-based data generation, and adversarial training, the proposed IDS method demonstrates enhanced performance in effectively identifying and mitigating intrusion attempts compared to classifier trained with existing datasamples.

5 Results and Discussion

In this research, we utilized Apache Spark and its MLlib library to conduct experiments and analyze results. We employed PySpark on Google Colab for distributed computing, conducting research on an M2 Chip Macbook, optimizing computational efficiency and applying distributed computing resources effectively.

We employed four well-known machine learning algorithms: Logistic Regression, k-nearest neighbors, Gaussian Mixture Models, Support Vector Machines (SVM), Random Forest (RF) as well as RF based GANs in this work. To assess the performance of the model, we relied on four key classification metrics: True Positives (correctly classified attacks), True Negatives (correctly classified normal records), False Positives (misclassified attacks), and False Negatives (misclassified normal records).

The primary performance metrics are given which are ratio of classified records to the total records, whether classified correctly or incorrectly. It is calculated as:

5.1 Accuracy Metrics

The Accuracy is calculated by the following equation:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

Precision is about how accurate your model is out of those predicted positive:

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

Recall is a measure of the proportion of real positive examples that a classifier can properly identify in which a greater recall indicates that fewer genuine positives were missed. Recall is calculated by the following equation:

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

The F1_score is used to express performance of algorithm which are between from 0 to 1. A value of 0 and 1 indicate poor performance, and perfect performance respectively and it can be described as a function of Precision and Recall:

$$F1Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{6}$$

This metric helps us evaluate the effectiveness of our intrusion detection system in accurately identifying network attacks and normal records.

5.2 Comparative Analysis with Existing Research

We implemented to generate plots to observed the stability of loss values for Generator and Discriminator while training with real and synthetic datasets for NSL-KDD dataset

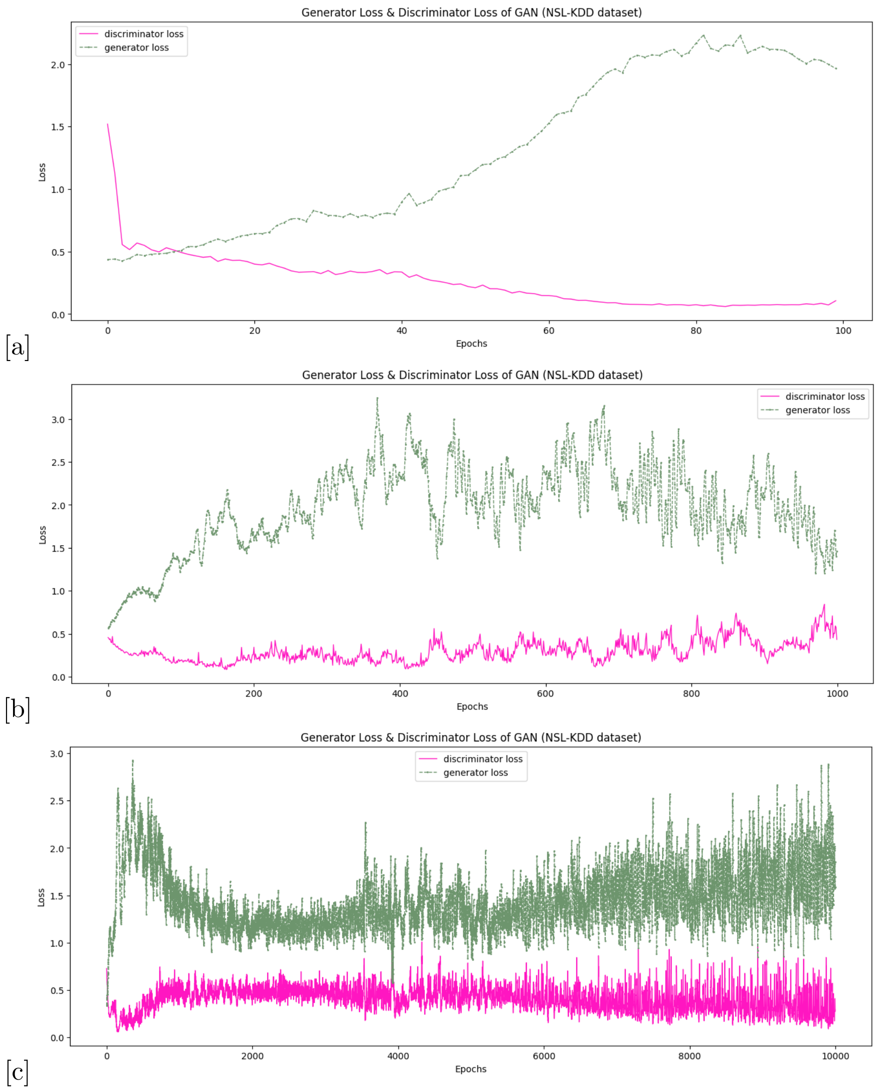

Fig. 4. Loss functions of Generator and Discriminator while training with NSL-KDD dataset by GANs for: (a) 100 epochs (b) 1000 epochs (c) 10000 epochs.

for different epochs. Also, there are the number of training epochs, and the minimum and maximum loss values recorded during training. For the NSL-KDD dataset, a minimum loss is 0.32 for the generator and a maximum loss is 2.9 for discriminator, while a minimum loss and maximum losses are 0.1 and 1.2 respectively (Fig. 4). Also, we observed that losses differences were seen between the epochs 2000 and 7000. However, the GAN model struggled more with the complexity and variability of various datasets, resulting in higher fluctuations and potentially slower convergence during training than quantum computing [18].

In our extensive exploration of machine learning models for intrusion detection, various algorithms were rigorously evaluated based on essential performance metrics. Logistic Regression demonstrated a commendable accuracy of 87.88%, showcasing its capacity to correctly identify both true positives (TP) and true negatives (TN). In K-Nearest Neighbors (KNN), the number of nearest neighbors is 2 for distinguishing normal from attacks as

Table 1 Performance Metrics for Different Models

Model	TP	TN	FP	FN	Accuracy
Logistic Regression	11259	10818	919	2127	0.8788
KNN	13275	11652	157	111	0.9894
Gaussian Naive Bayes	12535	10545	1264	851	0.9161
SVM	13099	11413	396	287	0.9729
RF	13374	11787	22	12	0.9987
GAN based RF	13378	11789	17	11	0.9994

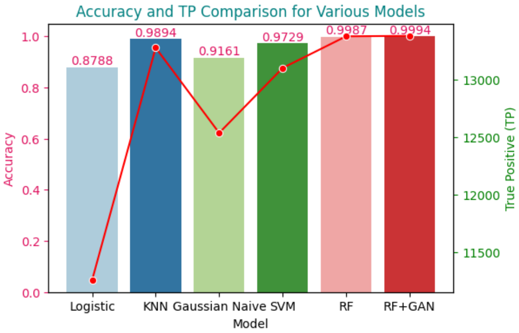

Fig. 5 Compare accuracy and true positives of various models

well as &4 for selecting the atatck type. It allows the classifier to consider the classification based on the five nearest neighbors that makes a balance between capturing local patterns and avoiding overfitting effectively. It excelled with an impressive accuracy of 98.94%, minimizing false positives (FP) and false negatives (FN). Gaussian Naive Bayes maintained a balanced performance, achieving an accuracy of 91.61% (Table 1). Support Vector Machine (SVM) exhibited a high accuracy of 97.29%, indicating its effectiveness in distinguishing between intrusions and normal instances (Fig. 5).

Table 2 Results Proposed work's and Existing Work

Authors	Algorithm	Acc. (%)	Dataset
Tang et al. [34]	DNN	75.75	NSL-KDD
Kim et al. [35]	LSTM	96.93	KDDCUP'99
Potluri et al. [36]	DL	97.50	NSL-KDD
Jiang et al. [37]	LSTM	98.94	NSL-KDD
Belouch et al. [38]	RF	97.00	UNSW-NB15
	DT	95.82	UNSW-NB15
	SVM	92.28	UNSW-NB15
	Naïve	74.19	UNSW-NB15
Rao et al., 2023 [39]	LSTM	98.97	UNSW-NB15
Proposed work	Gaussian Bayes	91.61	NSL-KDD
	SVM	97.29	NSL-KDD
	RF	99.87	NSL-KDD
	GAN based RF	**99.94**	**NSL-KDD**

In our comprehensive exploration of various machine learning techniques for intrusion detection, Tang et al. utilized Deep Neural Networks (DNN) achieving a 75.75% accuracy on the NSL-KDD dataset [34]. Long Short-Term Memory networks (LSTM) was used by Kim et al. for a notable 96.93% accuracy on the KDDCUP'99 dataset [35]. Moreover, Jiang et al. 2018 and Rao et al., 2023 used NSL-KDD and UNSW-NS15 datasets for training

LSTM model seperately which has accuracy 98.94% and 98.97% repectively [37, 39] (Table 2). However, the proposed work outperformed existing all works, presenting a novel application of Random Forest (RF) on both real and synthetic datasets generated by a Generative Adversarial Network (GAN). This hybrid RF+GAN model demonstrated exceptional performance, achieving an impressive 99.94% accuracy on the NSL-KDD dataset augmented with GAN-generated samples, showcasing the efficacy of our proposed methodology in handling imbalanced datasets for intrusion detection.

6 Conclusion

In conclusion, we have focused the utilization of Generative Adversarial Networks (GANs) in a potential ways for improving the performance of GAN based RF classifier for efficient intrusion detection systems (IDS). In order to address issues related to imbalanced data and missing samples, we applied GANs for the generation of synthetic data, thereby augmenting limited datasets and enhancing the training process. In this paper, the proposed GAN-based Random Forest (RF) model demonstrates the advancements in defense techniques for cybersecurity in distributed computing so that we can handle huge amount of data using clusters of machine in PySpark. Through evaluation, it was observed that integrating synthetic data with existing datasets significantly enhances model stability and robustness, resulting in a much improvement in the detection rate of cyber attacks as it achieved an accuracy 99.94% which outperforms the existing works. This work showed the promising capacity of utilizing GANs to strengthen IDS effectiveness, emphasizing the need for handling and exploring new adversarial networks in this domain to enhance cybersecurity protocols.

Acknowledgements

The work is supported by the National Science Foundation under NSF Award 1946442, 2433800. Any opinions, findings, recommendations, expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] Grahn, K., Westerlund, M., Pulkkis, G. (2017). "Analytics for Network Security: A Survey and Taxonomy." In *Information Fusion for Cyber-Security Analytics*, Volume 691. Springer, pp. 175–193. doi:10.1007/978-3-319-44257-0-8.

[2] Chen, Q., Bridges, R. A. (2017). "Automated Behavioral Analysis of Malware: A Case Study of WannaCry Ransomware." In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, Cancun, Mexico, pp. 454–460. doi:10.1109/ICMLA.2017.0-119.

[3] Chalapathy, R., Chawla, S. (2019). "Deep Learning for Anomaly Detection: A Survey." arXiv preprint arXiv:1901.03407, pp. 1–50.

[4] Shahriar, H., Zulkernine, M. (2012). "Mitigating Program Security Vulnerabilities: Approaches and Challenges." *ACM Computing Surveys (CSUR)*, 44(3), pp. 1-46.

[5] Shahriar, H., Weldemariam, K., Zulkernine, M., Lutellier, T. (2014). "Effective Detection of Vulnerable and Malicious Browser Extensions." *Computers Security*, 47, pp. 66-84.

[6] Azad, M. A. K., Khatun, A., Rahman, M. A. (2017). "A Slotted-Sense Streaming MAC for Real-Time Multimedia Data Transmission in Industrial Wireless Sensor Networks." *International Journal of Advanced Engineering Research and Science*, 4(3).

[7] Wang, Y., Wong, J., Miner, A. (2004). "Anomaly Intrusion Detection Using One Class SVM." In *Proceedings from the Fifth Annual IEEE SMC Information Assurance Workshop*. IEEE, West Point, NY, USA, pp. 358–364. doi:10.1109/IAW.2004.1437839.

[8] Hasan, M. A. M., Nasser, M., Pal, B., Ahmad, S. (2014). "Support Vector Machine and Random Forest Modeling for Intrusion Detection System (IDS)." *Journal of Intelligent Learning Systems and Applications*, 6(1), pp. 45. doi:10.4236/jilsa.2014.61005.

[9] Deng, H., Zeng, Q.-A., Agrawal, D. P. (2003). ”SVM-Based Intrusion Detection System for Wireless Ad Hoc Networks.” In *2003 IEEE 58th Vehicular Technology Conference. VTC 2003-Fall.* IEEE, Orlando, FL, USA, Volume 3, pp. 2147–2151. doi:10.1109/VETECF.2003.1285404.

[10] Amiri, F., Yousefi, M. R., Lucas, C., Shakery, A., Yazdani, N. (2011). ”Mutual Information-Based Feature Selection for Intrusion Detection Systems.” *Journal of Network and Computer Applications*, 34(4), pp. 1184–1199. doi:10.1016/j.jnca.2011.01.002.

[11] Ambusaidi, M. A., He, X., Nanda, P., Tan, Z. (2016). ”Building an Intrusion Detection System Using a Filter-Based Feature Selection Algorithm.” *IEEE Transactions on Computers*, 65(10), pp. 2986–2998. doi:10.1109/TC.2016.2519914.

[12] Rahman, M. A. (2020). ”Detection of Distributed Denial of Service Attacks Based on Machine Learning Algorithms.” *International Journal of Smart Home*, 14(2), pp. 15-24.

[13] Amor, N. B., Benferhat, S., Elouedi, Z. (2004). ”Naive Bayes vs Decision Trees in Intrusion Detection Systems.” In *Proceedings of the 2004 ACM Symposium on Applied Computing.* ACM, Nicosia, Cyprus, pp. 420–424. doi:10.1145/967900.967989.

[14] Panda, M., Patra, M. R. (2007). ”Network Intrusion Detection Using Naive Bayes.” *International Journal of Computer Science and Network Security*, 7(12), pp. 258–263.

[15] Zhang, J., Zulkernine, M., Haque, A. (2008). ”Random-Forests-Based Network Intrusion Detection Systems.” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(5), pp. 649–659. doi:10.1109/TSMCC.2008.923876.

[16] Rahman, Md Abdur, and Shahriar Hossain. (2023, December). “Clustering Enabled Robust Intrusion Detection System for Big Data using Hadoop-PySpark”, 2023 IEEE 20th International Conference on Smart Communities: Improving Quality of Life using AI, Robotics and IoT (HONET), Boca Raton, Florida, USA.

[17] Akter, Shapna, Rahman, Md. Abdur, Shahriar Hossain, Rahman, Muhammad. (2023, December). Early Prediction of Cryptocurrency Price Decline: A Deep Learning Approach. 26th International Conference on Computer and Information Technology (ICCIT), Cox’s Bazar, Bangladesh (accepted and presented).

[18] Rahman, Md Abdur, Hossain Shahriar, Victor Clincy, Md Faruque Hossain, and Muhammad Rahman. “A Quantum Generative Adversarial Network-based Intrusion Detection System.” In 2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC), pp. 1810-1815. IEEE, 2023. [18-22]

[19] A. Chawla, B. Lee, S. Fallon, P. Jacob, Host based intrusion detection system with combined CNN/RNN model, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, Dublin, Ireland, 2018, pp. 149–158, doi:10.1007/978-3-030-13453-2-12.

[20] M. Moradi, M. Zulkernine, A neural network based system for intrusion detection and classification of attacks, in: Proceedings of the IEEE International Conference on Advances in Intelligent Systems - Theory and Applications, IEEE, Luxembourg-Kirchberg, Luxembourg, 2004, pp. 15–18.

[21] Z. Li, Z. Qin, K. Huang, X. Yang, S. Ye, Intrusion detection using convolutional neural networks for representation learning, in: International Conference on Neural Information Processing, Springer, Guangzhou, China, 2017, pp. 858–866, doi:10.1007/978-3-319-70139-4-87.

[22] S. Aljawarneh, M. Aldwairi, M.B. Yassein, Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model, J. Comput. Sci. 25 (2018) 152–160, doi:10.1016/j.jocs.2017.03.006.

[23] Y. Yang, K. Zheng, C. Wu, Y. Yang, Improving the classification effectiveness of intrusion detection by using improved conditional variational autoencoder and deep neural network, Sensors 19 (11) (2019) 2528, doi:10.3390/s19112528.

[24] Y. Hong, U. Hwang, J. Yoo, S. Yoon, How generative adversarial networks and their variants work: an overview, ACM Comput. Surv. (CSUR) 52 (1) (2019) 10, doi:10.1145/3301282.

[25] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, D.N. Metaxas, StackGAN: text to photo-realistic image synthesis with stacked generative adversarial networks, in: Proceedings of the IEEE International Conference on Computer Vision, IEEE, Venice, Italy, 2017, pp. 5907–5915.

[26] K. Lei, M. Qin, B. Bai, G. Zhang, M. Yang, GCN-GAN: a non-linear temporal link prediction model for weighted dynamic networks, in: IEEE INFOCOM 2019- IEEE Conference on Computer Communications, IEEE, Paris, France, 2019, pp. 388–396, doi:10.1109/INFOCOM.2019.8737631.

[27] L. Yu, W. Zhang, J. Wang, Y. Yu, SeqGAN: Sequence generative adversarial nets with policy gradient, in: Thirty-First AAAI Conference on Artificial Intelligence, AAAI, San Francisco, California, USA, 2017, pp. 1–7.

[28] H. Shi, J. Dong, W. Wang, Y. Qian, X. Zhang, SSGAN: secure steganography based on generative adversarial networks, in: 18th Pacific-Rim Conference on Multimedia, volume 1, Springer, Harbin, China, 2017, pp. 534–544, doi:10.1007/978-3-319-77380-3-51.

[29] J. Gauthier, Conditional generative adversarial nets for convolutional face generation, Class Project Stanford CS231N 2014 (5) (2014) 2.

[30] G. Douzas, F. Bacao, Effective data generation for imbalanced learning using conditional generative adversarial networks, Expert Syst. Appl. 91 (2018) 464–471, doi:10.1016/j.eswa.2017.09.030.

[31] Vu, L., Bui, C.T., Nguyen, Q.U. (2017). A deep learning based method for handling imbalanced problem in network traffic classification. In Proceedings of the Eighth International Symposium on Information and Communication Technology, SoICT 2017, ACM, Nha Trang City, Viet Nam, pp. 333–339. doi:10.1145/3155133.3155175.

[32] Mbow, M., Koide, H., Sakurai, K. (2021, November). An intrusion detection system for imbalanced dataset based on deep learning. In 2021 Ninth International Symposium on Computing and Networking (CANDAR) (pp. 38-47). IEEE.

[33] Fan, W., Bifet, A. (2013). Mining big data: Current status and forecast to the future. ACM SIGKDD Explorations Newsletter, 14(2), 1–5.

[34] Tang, T. A., Mhamdi, L., McLernon, D., Zaidi, S. A. R., Ghogho, M. (2016). Deep learning approach for network intrusion detection in software-defined networking. In Proceedings of the 2016 International Conference on Wireless Networks and Mobile Communications, Fez, Morocco, pp. 258–263.

[35] Kim, J., Kim, J., Thu, H. L. T., Kim, H. (2016). Long short-term memory recurrent neural network classifier for intrusion detection. In Proceedings of the 2016 International Conference on Platform Technology and Service, Jeju, Republic of Korea, pp. 1–5.

[36] Potluri, S., Diedrich, C. (2016). Accelerated deep neural networks for an enhanced intrusion detection system. In Proceedings of the 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation, Berlin, Germany, pp. 1–8.

[37] Jiang, F., Fu, Y., Gupta, B. B., Liang, Y., Rho, S., Lou, F., Meng, F., Tian, Z. (2018). Deep learning-based multichannel intelligent attack detection for data security. IEEE Transactions on Sustainable Computing, 5(2), 204–212.

[38] Belouch, M., El Hadaj, S., Idhammad, M. (2018). Performance evaluation of intrusion detection based on machine learning using Apache Spark. Procedia Computer Science, 127, 1-6.

[39] Rao, Yamarthi Narasimha, and Kunda Suresh Babu. "An imbalanced generative adversarial network-based approach for network intrusion detection in an imbalanced dataset." Sensors 23.1 (2023): 550.