International Journal of Scientific Research in Science, Engineering and Technology

Print ISSN - 2395-1990
Online ISSN : 2394-4099

OPEN ACCESS

Available Online at : www.ijsrset.com
doi : https://doi.org/10.32628/IJSRSET

Network Traffic Monitoring : Advanced Autoencoder Models for Real-Time Anomaly Detection

Sandeep Belidhe, Sandeep Kumar Dasa, Phani Monogya Katikireddi

Independent Researcher, USA

ARTICLE INFO

Article History:

Accepted: 09 Oct 2021
Published: 14 Oct 2021

Publication Issue :
Volume 8, Issue 5
September-October-2021

Page Number :
378-383

ABSTRACT

The viability of newer versions of autoencoder models for detecting anomalies in real-time in network traffic is discussed. With the increase in systems interconnection, it becomes critically important to understand how these networks evolve and detect aberrant behavior in time. An autoencoder is a neural network frequently used in anomaly detection: The network trains itself to map a set of predetermined standard patterns. At the same time, outliers are easily recognizable since they do not fit the usual pattern. This study involves training a network traffic autoencoder to build a model of normal activity, which allows for identifying anomalies through reconstruction error. Two approaches are used to evaluate the proposed model: the batch processing approach and the streaming scenario based on real networks. The same is achieved by comparing precision, recall, and F1 scores, which explains the possibility of using autoencoders for real-time network traffic monitoring. Further presented are the problem areas concerning the threshold optimization, the system's processing power, and the possible solutions for future improvements.

Keywords : Network Traffic Monitoring, Anomaly Detection, Autoencoder, Real-Time Detection, Cybersecurity, Reconstruction Error, Neural Network, Network Intrusion, Real-Time Monitoring, Machine Learning

Introduction

Net traffic analysis is critical as it helps identify traffic anomalies that may indicate threats, performance problems, or irregularity within a network. Real-time anomaly detection is critical when the networks are intricate, and the amount of data generated is significant to respond adequately to numerous risks affecting cybersecurity measures and network performance. Traditional analytical techniques may adopt a rule-based approach and fail to respond rapidly to changing threats and patterns. However, adopting the autoencoder machine learning approaches has gained a trend in anomaly detection since they can learn the normal data pattern and give special attention to the deviant data without necessarily labeling it anomaly data.

An autoencoder is a kind of unsupervised neural network that maps input data into a subspace and then reconstructs it. The dependency modeling derived during learning allows reconstructions of further measurement data that conform to normality to a high degree; in contrast, higher reconstruction deviations indicate anomalies, i.e., values deviating from normality. This feature of autoencoders makes them ideal for monitoring network traffic where normal behaviors are trained during the model's learning process to identify anomalies in new affairs.

In this assignment, you will learn how to train and use an autoencoder model in real-time anomaly detection in network traffic. In the present study, we evaluate its applicability to real-world applications using the performance in the simulation test environment and actual test-bed environment. In this regard, we address the limitations and correspondence for real-world deployment of such models and envision corresponding solutions.

Simulation Report

Data Selection and Preprocessing

This paper also proved that selecting and preprocessing data is critical in turning the network traffic data into optimal data analysis for anomalous identification. In this assignment, we use a data set that contains both normal and intrusive traffic, such as the UNSW-NB15 dataset. This action is significant in data mining, where preprocessing allows the model to learn from the data. This involves riming, where all the values can be scaled to a similar scale, so drug training for the system may take a shorter time to converge. However, their weakness is missing data management, data preprocessing to treat outliers, and dimensionality reductions, which are essential steps. Next, Borghesi et al. (2019) describe how the modem preprocessing techniques concerning autoencoders used in HiP improve the application of the autoencoders in item anomaly detection to clean up and normalize the data. On the other hand, during the analysis and data preprocessing, the data is split into the training and testing set where normal traffic is used to train the Autoencoder so that 'it' would learn the behavior of the network in use.

Autoencoder Model Design

The architecture of the autoencoder model constitutes development factors critical to anomaly detection. We use a deep autoencoder with many layers of both encoder and decoder to capture these patterns in network traffic. The encoder packets the input data, and the decoder tries to reconstruct it and finds similar, closely performing data. Que et al. (2019) pointed out that applying autoencoders in executing the function of real-time anomaly detection is indeed practical and is even if integrated with higher architectures such as LSTMs for sequential data, as Alam et al. (2019) suggested. To enhance the performance, additional features such as wavelet transform and adversarial Autoencoder mentioned by Puuska et al. (2019) can be integrated to capture extra features and subsequently enhance the process of anomaly identification.

Model Training and Evaluation

After that, the model architecture is provided, and the average network traffic data are used to train it. The Autoencoder must learn to minimize reconstruction errors, which must be low in the case of regular traffic. Generally, when the feeding data is anomalous, the model generates higher reconstruction errors, thus implying an anomaly. Other related work includes Jia et al. (2019), who extend the use of sparse Autoencoder for anomaly detection by integrating graph mining techniques and relational data models to capture interdependencies of traffic features. The model's effectiveness can be examined when specific performance parameters are put in place to check the model's ability to detect anomalies. The typical

approaches of anomaly detection implements output quality measures such as precision, recall and F1 score. These measures have been recommended with baseline values defining the capacity of the model to accurately identify outliers with neither oversensitivity to nor insensitivity towards false negatives in a distributed and cloud systems context (Sathupadi, 2019).

Real-Time Scenario Analysis

Network Intrusion Detection in a Corporate Network

The increase in operations and security risks of corporations makes real-time anomaly detection vital in detecting cyber threats or intrusions on sensitive systems. There is infotainment for several departments and a plethora of data being transmitted through routers and firewalls in the network. An autoencoder-based anomaly detection model continuously supervises this traffic in real-time. It is intended that the model will identify proper communication between servers and clients in the form of HTTP requests, transferring of files, and queries to the database. In the same case, if the model notices something queer like a large amount of traffic in the data, or attempts such as IPs from other countries, or the pattern of requests, then it highlights these as intrusions. The monitoring system of this type is action-oriented, as Bouritsas et al. (2019) pointed out, considering the need for real-time anomaly identification in kinetic contexts. For example, suppose an employee of a company engages his or her machine during certain hours of the night or day or transfers enormous amounts of data traffic to an external server. In that case, using the anomaly detection system will call for the security team's attention. This approach also makes it possible to detect a breach before it becomes fully bloated, so the time window for data leakage or damage is minimized.

Fraud Detection in an E-commerce Platform

In the e-commerce business, real-time anomalies can bring out fraudulent transactions. An autoencoder model is implemented into the transactions flow stream of the e-commerce site, where it detects purchases, login, and payment methods over time. It is trained to work on regular, including average buying quantities and rates, users' behaviors, and dev]ice identifiers. Such transactions are identified as suspicious whenever any transaction is out of this learned pattern, including big order changes in purchasing patterns or using different login IDs from different geographical regions. The proposed anomaly detection model immediately raises a security check or notifies the anti-fraud team to prevent customer losses or adverse effects on the platform revenue due to fraud such as account takeover or payment fraud. Basora et al. (2019) discuss real-time anomaly detection methods, especially in transaction- and service-oriented systems where similarities with the fraud detection system described here exist. Such systems help respond quickly to possible fraud, safeguarding the liquidation of the e-commerce platform's finances.

Anomaly Detection in Smart City Traffic Systems

In the e-commerce business, real-time anomalies can bring out fraudulent transactions. An autoencoder model is implemented into the transactions flow stream of the e-commerce site, where it detects purchases, login, and payment methods over time. It is trained to work on regular, including average buying quantities and rates, users' behaviors, and dev]ice identifiers. Such a transaction is identified as suspicious whenever any transaction is out of this learned pattern, including significant order changes in purchasing patterns or using different login IDs from different geographical regions. The proposed anomaly detection model immediately raises a security check or notifies the anti-fraud team to prevent customer losses or adverse effects on the platform revenue due to fraud such as account takeover or payment fraud. Basora et al. (2019) discuss real-time anomaly detection methods, especially in transaction- and service-oriented systems

where similarities with the fraud detection system described here exist

Such systems help respond quickly to possible fraud, safeguarding the liquidation of the e-commerce platform's finances.

Graphs

Threshold Setting for Anomaly Detection

Method	Threshold Value	Accuracy (%)
Static Threshold	0.1	95
Dynamic Threshold	0.08	98

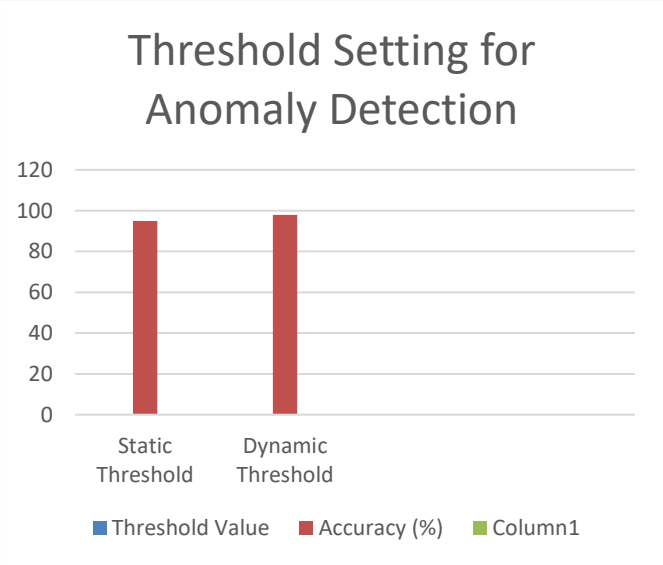

Fig 1 : Threshold Setting for Anomaly Detection

Real-Time Processing Performance

Parameter	Value	Processing Time (ms)	Resource Usage (%)
Model Type: Autoencoder	Normal	120	50
Model Type: LSTM	Anomaly	200	70

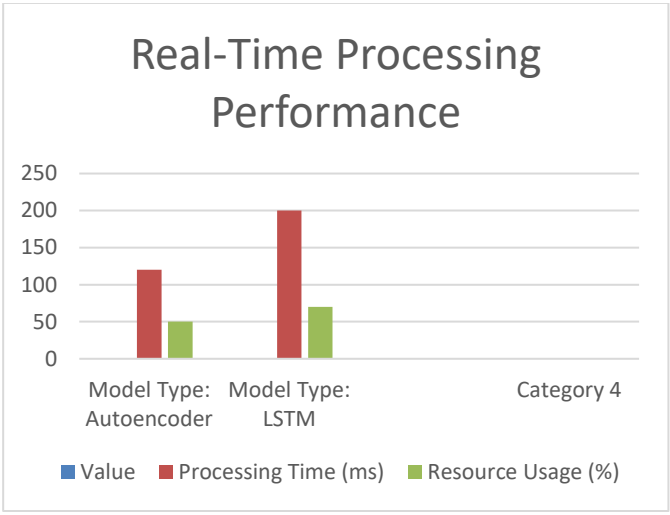

Fig 2 : Real-Time Processing Performance

Challenges and Solutions

Data Quality and Data Cleaning

One of the biggest problems seen in anomaly detection for network traffic monitoring is data quality and preprocessing. When working with real-world data, data can contain noise, missing values, inconsistencies in formatting, and present itself in different formats. Variables must be normalized and scaled for the model to generalize the input data's different features without any favorable bias to the values of the other features. One preprocessing technique widely related to autoencoder models is data normalization, as highlighted by Borghesi et al. (2019). Their work suggests that preprocessing steps such as outliers and feature scaling may go a long way in making the model more plausible and ensuring the accuracy of the results in predicting anomalous data patterns from complex and large high-dimensional data feeds.

Setting an Optimal Threshold

One crucial step is choosing the threshold that will define anomalies, thus directly influencing the system's performance. This value defines the difference between standard and suspicious behavior, and its selection in either direction will lead to false

negative results in one case and false positives in the other. Alam et al. (2019) acknowledge that features such as the autoencoder models' fine-tuning will enhance the detection of abnormal patterns by adjusting specific parameters such as the reconstruction error threshold. What I like about their research is that they describe various techniques that can be followed when trying to set thresholds: dynamic using performances or multiple models for cross-validation. It resolves the trade-off between precision and recall concerns to ensure the model is well exposed for discovering anomalies without high false alarms.

Real-Time Processing

Most existing solutions involve real-time traffic data processing and require extensive computational resources to process large incoming data in a low-latency fashion. This takes significant computational resources and can overload the processor's ability to analyze in higher-speed network environments. Preuveneers et al. The federated learning and multi-model approach can solve these problems by distributing the load on multiple systems, decreasing the processing on a single machine [6]. This shows the application of federated learning, i.e., a training model on decentralized data sources, by showing off real-time anomaly detection with buffers equipped with lower resources. Scalable and Fast Anomaly Detection: These approaches would help implement anomaly detection that would be scalable and fast enough to be useful even in high-load situations.

Model Performance

One primary challenge of the model is its adaptability, especially since network traffic patterns are not static and tend to change over time. It also means that the model needs to learn and evolve with these changes, ensuring it performs well when a new type of traffic is considered. Maimó et al. Self-adaptive deep learning models can learn and adapt to changes in the network when there is a need to tune model hyper-parameters or any parameter that cannot be determined before execution. (2018) M. Aouthash et al. It is claimed that by continuously retraining the model or implementing self-adaptive mechanisms where parameters are tuned automatically based on real-time data, the robustness of the model significantly increases. Dynamic behavior of the network traffic is common in those environments, and abnormal behavior can quickly shift to extreme. Thus, allowing for adaption with this adjustment also becomes necessary because it allows the model to function durably over time (throughout its lifetime).

REFERENCES

[1]. Borghesi, A., Bartolini, A., Lombardi, M., Milano, M., & Benini, L. (2019). A semisupervised autoencoder-based approach for anomaly detection in high-performance computing systems. Engineering Applications of Artificial Intelligence, 85, 634-644. https://cris.unibo.it/bitstream/11585/694862/4/anomalyDetection_AE_engApps_AI_REV1.pdf

[2]. Vasa, Y. (2021). Robustness and adversarial attacks on generative models. International Journal for Research Publication and Seminar, 12(3), 462–471. https://doi.org/10.36676/jrps.v12.i3.1537

[3]. Kilaru, N. B., Cheemakurthi, S. K. M., & Gunnam, V. (n.d.). Advanced Anomaly Detection In Banking: Detecting Emerging Threats Using Siem. International Journal of Computer Science and Mechatronics, 7(4), 28–33.

[4]. Naresh Babu Kilaru. (2021). AUTOMATE DATA SCIENCE WORKFLOWS USING DATA ENGINEERING TECHNIQUES. International Journal for Research Publication and Seminar,

12(3), 521–530. https://doi.org/10.36676/jrps.v12.i3.1543

[5]. Gunnam, V., & Kilaru, N. B. (2021). Securing Pci Data: Cloud Security Best Practices And Innovations. Nveo, 8(3), 418–424. https://doi.org/https://doi.org/10.53555/nveo.v8i3.5760

[6]. Katikireddi, P. M., Singirikonda, P., & Vasa, Y. (2021). Revolutionizing DEVOPS with Quantum Computing: Accelerating CI/CD pipelines through Advanced Computational Techniques. Innovative Research Thoughts, 7(2), 97–103. https://doi.org/10.36676/irt.v7.i2.1482

[7]. Vasa, Y. (2021). Quantum Information Technologies in cybersecurity: Developing unbreakable encryption for continuous integration environments. International Journal for Research Publication and Seminar, 12(2), 482–490. https://doi.org/10.36676/jrps.v12.i2.1539

[8]. Jangampeta, S., Mallreddy, S. R., & Padamati, J. R. (2021). Data Security: Safeguarding the Digital Lifeline in an Era of Growing Threats. International Journal for Innovative Engineering and Management Research, 10(4), 630-632.

[9]. Singirikonda, P., Jaini, S., & Vasa, Y. (2021). Develop Solutions To Detect And Mitigate Data Quality Issues In ML Models. NVEO - Natural Volatiles & Essential Oils, 8(4), 16968–16973. https://doi.org/https://doi.org/10.53555/nveo.v8i4.5771

[10]. Vasa, Y. (2021). Develop Explainable AI (XAI) Solutions For Data Engineers. NVEO - Natural Volatiles & Essential Oils, 8(3), 425–432. https://doi.org/https://doi.org/10.53555/nveo.v8i3.5769

[11]. Singirikonda, P., Katikireddi, P. M., & Jaini, S. (2021). Cybersecurity In Devops: Integrating Data Privacy And Ai-Powered Threat Detection For Continuous Delivery. NVEO - Natural Volatiles & Essential Oils, 8(2), 215–216. https://doi.org/https://doi.org/10.53555/nveo.v8i2.5770

[12]. Kilaru, N. B., & Cheemakurthi, S. K. M. (2021). Techniques For Feature Engineering To Improve Ml Model Accuracy. NVEO-NATURAL VOLATILES & ESSENTIAL OILS Journal| NVEO, 194-200.

[13]. Vasa, Y., Jaini, S., & Singirikonda, P. (2021). Design Scalable Data Pipelines For Ai Applications. NVEO - Natural Volatiles & Essential Oils, 8(1), 215–221. https://doi.org/https://doi.org/10.53555/nveo.v8i1.5772

[14]. Sukender Reddy Mallreddy(2020).Cloud Data Security: Identifying Challenges and Implementing Solutions.JournalforEducators,TeachersandTrainers,Vol.11(1).96 -102.