Deep Learning Enabled Network Intrusion Systems

Justin Wasser

University of Maryland Global Campus

INFA 620: Network and Internet Security

Professor Crisan

3/27/2023

Contents

Introduction

The following research paper examines the current practices of network traffic analysis, including the issues it faces, and the emerging technologies that will drive its evolution in the near future. Topics analyzed include an overview of current standard network traffic monitoring practices, their purpose, how they work, and their limitations. Furthermore, an exploration of the changing nature (IoT) and size of networks will be conducted, including how these developments will likely impact current traffic analysis methods. Moreover, these novel network types necessitate a comparison between machine learning (ML), artificial neural networks (ANNs), and deep learning (DL). Lastly, deep learning-enabled network traffic intrusion tools will be examined, and a conclusion will be reached about the role this technology will play in the future of network traffic analysis.

Network Traffic Analysis Overview

The purpose of network traffic analysis at its core is to help preserve the confidentiality, integrity, and availability (CIA) of an information network in order for an organization to reliably use that information network to achieve its business goals (Chai, 2023; English, 2020). To that point, network traffic analysis is an important part of sound information and network security architecture as it is used to identify potential threats and attacks early on. In this way, traffic analysis allows an organization to limit the harm caused by those threats and attacks, and therefore helps preserve the confidentiality and integrity of its information network (Chai, 2023; English, 2020). Moreover, the other main aspect of network traffic analysis involves using tools that evaluate the availability of a network in detail, going well beyond just a simple 'yes or no' regarding network availability (English, 2020). These network details include measuring "network throughput, packet loss ratio, latency, and jitter (or delay variation)" (Abbasi et al., 2021). However, for the purposes of this essay, the focus will center on how network traffic analysis can be used to secure a network as opposed to maximizing its efficiency.

Moreover, whether analyzing network traffic for security or performance purposes, certain essential network traffic analysis practices must be implemented first (English, 2020). The first of which requires creating "a baseline so you know what's normal" (English, 2020), otherwise you will not be able to understand what a slow network or an attack looks like as you will have no reference point from which to compare (English, 2020). Another essential traffic analysis practice is to understand how your network data is being gathered, as both active and passive traffic-gathering methods can be employed (Abbasi et al., 2021; English, 2020). To that point, active gathering procedures refer to "generating and injecting probe traffic into a network in order to learn about the state of the network" (Abbasi et al., 2021) while passive collection practices refer to analyzing naturally occurring network traffic (obtained by placing passive probes at network interfaces) after it has occurred (Abbasi et al., 2021). The benefit of active gathering is that it provides a controlled environment where specific traffic situations can be observed in real-time, thereby enabling a deeper level of insight into the traffic data (Abbasi et al., 2021). While the benefit of passive gathering is that it can be used to help analyze traffic after it has occurred, including most valuably, "in post-event situations" (Abbasi et al., 2021). Furthermore, post-incident analysis allows network administrators/analysts to gain insights into what led to certain network conditions, and as a result, they can better protect the network against similar situations occurring in the future (Abbasi et al., 2021). However, for analysis to occur, the collected network traffic data must first undergo a series of processes that enable insights to be drawn out of the raw data.

Network Traffic Analysis Methodology

Once traffic data has been collected from the appropriate network, it undergoes a four-part process consisting of set selection, preprocessing, analysis, and evaluation (Joshi & Hadi, 2015). First, data is compiled into a standardized set that is based on what type of analysis is required (Joshi & Hadi, 2015). For example, “KDD cup data” (Joshi & Hadi, 2015) is commonly used to assess intrusion detection events within a larger network traffic environment, while a “CAIDA data set” (Joshi & Hadi, 2015) is used to assess different types of denial-of-service attacks (Joshi & Hadi, 2015). Once a specific data set has been chosen, the next step in the network traffic analysis methodology is preprocessing the chosen data set (Joshi & Hadi, 2015).

Preprocessing involves removing any “incomplete or inconsistent” (Joshi & Hadi, 2015) data from a data set, the reason this is done is to improve the consistency and therefore quality of a data set (Joshi & Hadi, 2015). Furthermore, there are two different types of preprocessing methods that a data set can undergo, the “discretization method” (Joshi & Hadi, 2015) and the “feature selection method” (Joshi & Hadi, 2015). Discretization preprocessing has four different variations all of which are used to condense data sets by transforming “continuous attributes” (Joshi & Hadi, 2015) into nominal (countable/non-infinity) attributes by dividing the continuous attribute into sections and assigning each section a nominal/countable value (Joshi & Hadi, 2015). On the other hand, the “feature selection method” (Joshi & Hadi, 2015) is used to remove sections from the data set that are irrelevant or repetitive to the practice of data mining (Joshi & Hadi, 2015).

After preprocessing is completed the data set is ready to begin being analyzed using general data mining methods (each of which contains further subsets/algorithms) (Joshi & Hadi, 2015). The four most common general data mining methods are clustering, classification, hybrid, and association; and these mining methods are categorized by whether they require data to be labeled/structured (Joshi & Hadi, 2015). Furthermore, data mining methods that require labeled data are referred to as “supervised learning” (Joshi & Hadi, 2015), examples of such classification methods include “decision tree approaches and support vector machine (SVM)” (Joshi & Hadi, 2015). While data mining methods that do not require labeled data are referred to as “unsupervised learning” (Joshi & Hadi, 2015), with an example being “clustering” (Sydorenko, 2020). The difference between the two methods is that supervised learning requires more human input in the form of conducting significant “feature extraction” (Kavlakoglu, 2020) or labeling of an unstructured data set (Kavlakoglu, 2020).

Finally, only after a type of data set is chosen, preprocessed, and mined, can evaluations be reached regarding the effectiveness of the data mining method/algorithm employed (Joshi & Hadi, 2015). To that point, the effectiveness of a data mining algorithm is measured in terms of its precision and its “time cost” (Joshi & Hadi, 2015). While many different metrics can be used to evaluate a data mining algorithm’s precision, the most used of those metrics is an algorithm’s accuracy ratio (Joshi & Hadi, 2015). A data mining algorithm’s accuracy ratio adds all correctly identify normal “true negatives” (Joshi & Hadi, 2015) and malicious “true positives” (Joshi & Hadi, 2015) packets and divides that number by the rate of classification errors an algorithm makes, i.e., malicious packet identified as normal “false negatives” (Joshi & Hadi, 2015) and normal packets identified as malicious “false positive” (Joshi & Hadi, 2015). Moreover, data mining algorithms are also evaluated based on the extent of computing resources they require to operate, which is simply called “time cost” (Joshi & Hadi, 2015). Lastly, the data mining algorithm that provides a desirable combination of precision and computing efficiency can be leveraged to create a network intrusion system (Ahmetoglu & Das, 2022).

Network Intrusion Systems

Given a baseline of data about a network’s traffic and armed with an algorithm used to analyze network traffic, traffic analysis tools can provide semi-automated or fully automated network security functions (Gillis, 2020). Specifically, traffic analysis tools such as an “intrusion detection system (IDS)” (Lutkevich, 2021) or an “intrusion prevention system (IPS)” (Gillis, 2020) can respond to suspicious network activity (data) by altering a network administrator in the case of an IDS or by taking a preventative measure without human intervention in the case of an IPS (Gillis, 2020). Furthermore, in any network of significant size, these tools are essential to the practice of network security via traffic analysis, as it is not feasible for humans to effectively or efficiently replicate the services these systems provide.

However, these tools are not foolproof as “IDSes are prone to false alarms -- or false positives” (Lutkevich, 2021), and even more consequentially IDSes do not always catch malicious network activity which is known as a “false negative” (Lutkevich, 2021). False positives can also be extremely harmful to an IDS as they “can lead to administrators ignoring alerts” (English, 2020) which severely degrades the impact of the IDS (English, 2020). Moreover, IPSes deal with the same false positive, and negative issues but the effects of false positives are more severe with an IPS as a false positive from an IPS will result in denying network services to an authorized user (Gillis, 2020). While in the case of IPSes repeated false negatives will degrade network capabilities by consuming network resources (Cisco, 2009). Consequently, IPSes users are faced with a difficult choice as “it is almost impossible to completely eliminate false positives and negatives without severely degrading the effectiveness of the IPS or severely disrupting the computing infrastructure of an organization” (Cisco, 2009). Therefore, although ultimately beneficial, IDSes and IPSes have their limitations. Moreover, those limitations will be exacerbated by an increasing number of smaller and smaller devices interfacing with each other, known as the internet of things (IoT).

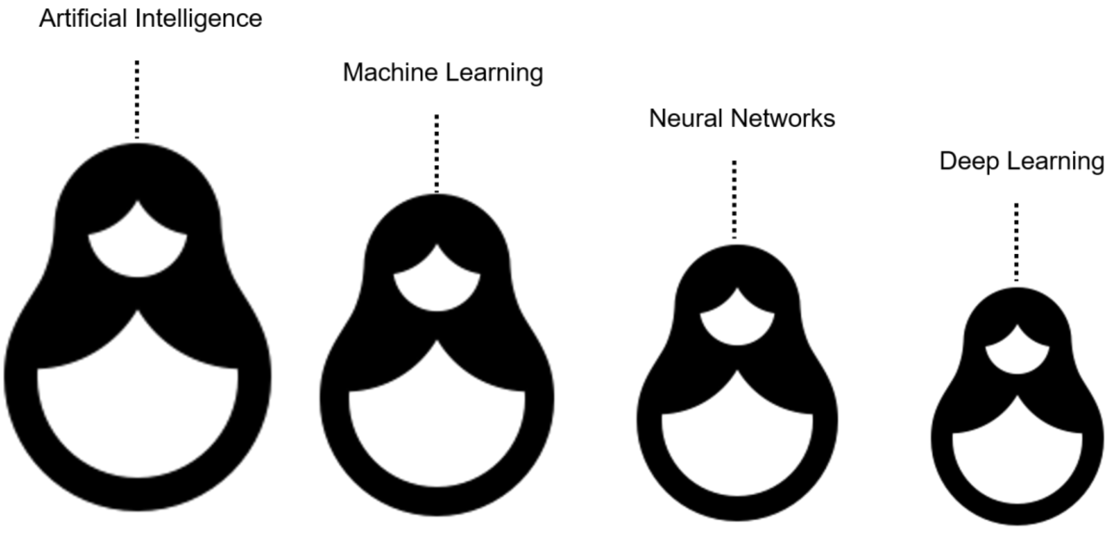
The Challenges IoT Networks Present to Machine Learning

IoT network traffic is rapidly expanding as exemplified by the fact that “the market for the Internet of Things is expected to grow 18% to 14.4 billion active connections” (Hasan, 2022) over the next year (Hasan, 2022). Therefore, it is not surprising that traditional traffic analysis methods may be pushed past their functional limit as the traffic they are tasked with analyzing rapidly expands (Hasan, 2022; Kavlakoglu, 2020). However, the size of the data that needs to be analyzed is not the only issue IoT networks present to ML-enabled data mining methods, as the type of data that IoT devices generally produce is of even greater consequence (Kavlakoglu, 2020). To that point, IoT network traffic is largely uncategorized, which (as explained in the earlier section covering data mining methods) for ML-rooted analysis, requires human supervision to analyze, and consequently is likely to be time/cost-prohibitive (Geetha & Thilagam, 2021; Kavlakoglu, 2020). This limitation presents a significant challenge to securing IoT networks as the growth and computing power of IoT devices means IoT networks “are a great target for attacks and malware” (Geetha & Thilagam, 2021), with “thousands of zero day attacks emerging in the field of Internet of things [IoT]” (Geetha & Thilagam, 2021). This reality only serves to heighten the imperative to secure said networks (Geetha & Thilagam, 2021).

This limitation of ML is significant because uncategorized (raw) data “will account for up to 80 percent of data by 2025” (Chen & Hormati, 2022), yet “traditional learning methods are poor in detection performance and accuracy” (Geetha & Thilagam, 2021). Therefore, there is a need for data mining methodologies that can accurately monitor IoT network traffic with less human input than ML requires (Abbasi et al., 2021).

Machine Learning vs Artificial Neural Networks vs Deep Learning

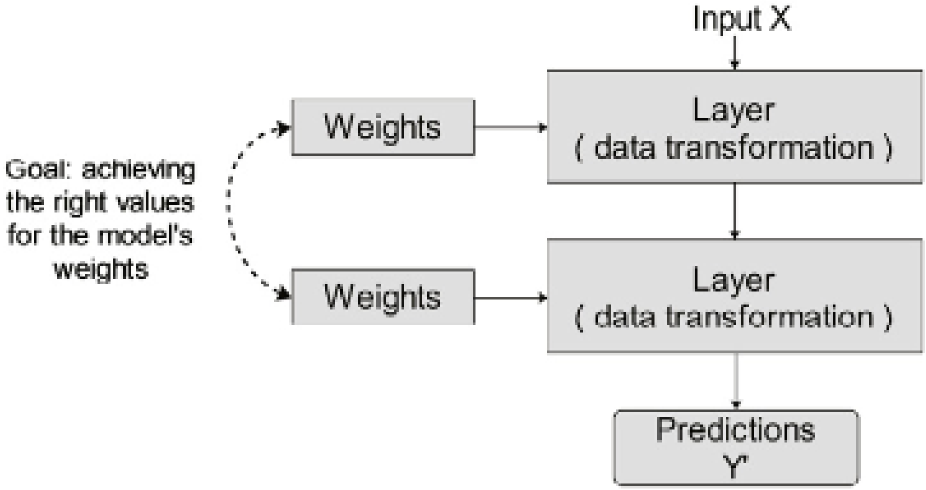
Deep learning (DL) is a subset of machine learning (ML), which separates itself from ML by its use of “artificial neural networks (ANNs)” (Kavlakoglu, 2020). Furthermore, ANNs are a subset of ML which lies in between ML and DL, which can be illustrated using the following “Russian nesting dolls” (Kavlakoglu, 2020) example (Kavlakoglu, 2020).



(Kavlakoglu, 2020)

ANN’s functionality can be described as attempting to “mimic the human brain through a set of algorithms” (Kavlakoglu, 2020). Additionally, ANNs are composed of four distinct sections which include, “inputs, weights, a bias or threshold, and an output” (Kavlakoglu, 2020). Lastly, ANNs consisting “of more than three layers” (Kavlakoglu, 2020) can be described as a DL where “the “deep” in deep learning is referring to the depth of layers” (Kavlakoglu, 2020).

DL’s multi-layered architecture is what separates it from ML and simple ANNs as it leverages its deep neural network to better recognize features (patterns) in unstructured/unlabeled data (Kavlakoglu, 2020). Moreover, the ‘depth’ of deep learning models can extend to “hundreds of successive layers of representations” (Abbasi et al., 2021). Furthermore, each successive layer has a multitude of weights or “parameters” (Abbasi et al., 2021) assigned that fundamentally change how analysis is performed based on their values (Abbasi et al., 2021). A diagram of how a single layer with accompanying weights functions is shown below.



(Abbasi et al., 2021)

After taking into consideration all the layers and their associated weights, some “DL models may have tens of millions of parameters” (Abbasi et al., 2021). This additional complexity means training a DL model requires a significantly larger data set and subsequently, significantly more computing power than an ML model requires (Kavlakoglu, 2020; Thompson et al., 2020). However, the benefit of employing a DL-based method is it requires less human input in the form of “feature extraction” (Kavlakoglu, 2020). Lastly, the amount of data produced by IoT-based network traffic and the infrastructure that supports IoT devices provide DL algorithms with both the data set size and computing resources it requires to operate (Abbasi et al., 2021; Kavlakoglu, 2020).

Deep Learning as a Solution to the IoT Network Challenge

Given enough time, computing power, and data points, deep learning (DL) can process raw (unstructured) data with minimal human-supervised instruction to produce accurate insights about the raw data using its deep artificial neural/belief network (Kavlakoglu, 2020). Therefore, given that the IoT produces a flood of unstructured data that would otherwise be unintelligible or require significant efforts by data scientists to structure, deep learning can be a very valuable tool for analyzing IoT network traffic (Hasan, 2022; Kavlakoglu, 2020).

The amalgamation of DL algorithms and IoT networks is possible due to the amount of data IoT networks produce (shown by its rapidly growing 12.2 billion connected devices as of 2021) which meets the high threshold required to train DL algorithms (Abbasi et al., 2021; Hasan, 2022). Additionally, new computing architectures had to be designed to support the IoT's growth, resulting in the creation/expansion of "Fog and Edge" (Abbasi et al., 2021) computing (Abbasi et al., 2021). Fog/Edge computing involves breaking up the large, centralized cloud computing infrastructure and distributing those separated smaller components closer to (Fog) or directly next to (Edge) where data is being generated (Haiston, 2022). The distributed "devices are equipped with high-performance computational equipment e.g. Graphical Processing Units (GPUs)" (Abbasi et al., 2021) which can be leveraged to provide the necessary computing power to run DL algorithms (Abbasi et al., 2021). Furthermore, Fog/Edge computing architecture makes it possible to use "distributed machine learning techniques" (Abbasi et al., 2021) to train DL algorithms, which means the DL algorithm can be broken up and trained closer to the data it is working with which lessens "the network overhead and jeopardization of security and privacy" (Abbasi et al., 2021).

Not only can deep learning be used to analyze IoT network traffic with less human input, but multiple tests have shown that it does so more accurately when compared to machine learning algorithms when tasked with identifying network attacks (Abbasi et al., 2021). For example, one test showed that a DL algorithm was more accurate when tasked with identifying "unknown attacks in MEC" (Abbasi et al., 2021) MEC meaning "Mobile Edge Computing" (Abbasi et al., 2021) than four different types of ML algorithms (Abbasi et al., 2021). Furthermore, another test of a novel DL algorithm "achieved an accuracy of 77.99%, compared to 59.71% accuracy acquired by the static technique" (Abbasi et al., 2021) with the static techniques including "classical ML" (Abbasi et al., 2021). Lastly, a study conducted on IoT network security methods found that "IoT malware can be detected using Recurrent Neural network (RNN) deep learning" (Geetha & Thilagam, 2021) and "by using RNN-IDS the accuracy and performance rate of intrusion detection is improved" (Geetha & Thilagam, 2021).

Conclusion

In conclusion, the number of connected devices and therefore data in transit (traffic) continues to grow rapidly due to the expansion of the IoT (Hasan, 2022). The growth in volume and complexity of IoT network traffic will reduce the efficiency and effectiveness of ML-enabled traffic analysis systems, in terms of their accuracy and the human capital they require to function. Therefore, deep learning needs to be utilized in novel network traffic technologies to resolve the challenges that IoT network traffic poses to existing ML-enabled traffic analysis systems. Furthermore, DL's ability to learn from uncategorized data with minimal human input, and potentially produce more accurate predictions is what makes it the essential component of the next generation of IDSes and IPSes (Fridman, 2019).

References:

Abbasi, M., Shahraki, A., & Taherkordi, A. (2021). Deep Learning for Network Traffic Monitoring and Analysis (NTMA): A Survey. *Computer Communications*, *170*, 19–41. https://doi.org/10.1016/j.comcom.2021.01.021

Ahmetoglu, H., & Das, R. (2022). A comprehensive review on detection of cyber-attacks: Data sets, methods, challenges, and future research directions. *Internet of Things*, *20*, 100615. https://doi.org/10.1016/j.iot.2022.100615

Chai, W. (2023, February). *What is the CIA Triad? Definition, Explanation and Examples*. WhatIs.com. https://www.techtarget.com/whatis/definition/Confidentiality-integrity-and-availability-CIA

Chen, C., & Hormati, A. (2022, October 20). *How to apply machine learning to unstructured data using BigQueryML*. Google Cloud Blog. https://cloud.google.com/blog/products/data-analytics/how-to-apply-machine-learning-to-unstructured-data-using-bigqueryml

Cisco. (2009, October 26). *Cisco Secure IPS - Excluding False Positive Alarms*. Cisco. https://www.cisco.com/c/en/us/support/docs/security/ips-4200-series-sensors/13876-f-pos.html

English, J. (2020, February). *Network traffic analysis best practices: Assess and repeat | TechTarget*. Networking. https://www.techtarget.com/searchnetworking/feature/Network-traffic-analysis-best-practices-Assess-and-repeat

Fridman, L. (2019). MIT Deep Learning Basics: Introduction and Overview [YouTube Video]. In *YouTube*. https://www.youtube.com/watch?v=O5xeyoRL95U

Geetha, R., & Thilagam, T. (2021). A Review on the Effectiveness of Machine Learning and Deep Learning Algorithms for Cyber Security. *Archives of Computational Methods in Engineering*, *28*(4), 2861–2879. https://doi-org.ezproxy.umgc.edu/10.1007/s11831-020-09478-2

Gillis, A. (2020, February). *What is an Intrusion Prevention System (IPS)?* SearchSecurity. https://www.techtarget.com/searchsecurity/definition/intrusion-prevention

Haiston, J. (2022, May 2). *Fog Computing vs. Edge Computing*. Symmetry Electronics. https://www.symmetryelectronics.com/blog/fog-computing-vs-edge-computing/#:~:text=With%20a%20name%20coined%20from

Harbert, T. (2021, February 1). *Tapping the power of unstructured data*. MIT Sloan. https://mitsloan.mit.edu/ideas-made-to-matter/tapping-power-unstructured-data

Hasan, M. (2022, May 22). *State of IoT 2021: Number of Connected IoT Devices Growing 9% to 12.3 Billion globally, Cellular IoT Now Surpassing 2 Billion*. IoT Analytics. https://iot-analytics.com/number-connected-iot-devices/

Joshi, M., & Hadi, T. (2015). *A Review of Network Traffic Analysis and Prediction Techniques*. https://arxiv.org/ftp/arxiv/papers/1507/1507.05722.pdf

Kavlakoglu, E. (2020, May 27). *AI vs. Machine Learning vs. Deep Learning vs. Neural Networks: What's the Difference?* IBM. https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks

Lutkevich, B. (2021, October). *What is an intrusion detection system (IDS)? Definition from SearchSecurity*. SearchSecurity. https://www.techtarget.com/searchsecurity/definition/intrusion-detection-system

Stedman, C., & Hughes, A. (2021, September). *What is Data Mining?* SearchBusinessAnalytics. https://www.techtarget.com/searchbusinessanalytics/definition/data-mining

Sydorenko, I. (2020, August 31). *Unlabeled Data in Machine Learning*. Labelyourdata.com. https://labelyourdata.com/articles/unlabeled-data-in-machine-learning#:~:text=However%2C%20unlabeled%20data%20can%20be

Thompson, N., Greenewald, K., Lee, K., & Manso, G. (2020). *THE COMPUTATIONAL LIMITS OF DEEP LEARNING*. https://ide.mit.edu/wp-content/uploads/2020/09/RBN.Thompson.pdf