Detecção de ataques em Internet das Coisas através de técnicas de aprendizado profundo

Vinicius Marino Luciano[1], **Bruno Bogaz Zarpelão**[1]

[1]Departamento de Computação – Universidade Estadual de Londrina (UEL)
Caixa Postal 10.011 – CEP 86057-970 – Londrina – PR – Brasil

vinicius.luciano@uel.br, brunozarpelao@uel.br

Abstract. *Internet of Things (IoT) is a global trend that is evolving and settling. Despite its potential, the propagation of IoT devices with little security embedded brings hard challenges to network security. Therefore, many proposals for intrusion detection systems have emerged to mitigate this problem. However, most of these works use supervised algorithms, which becomes a problem, as labeling the training data is a slow and difficult task. As a solution to this problem, this work proposes an intrusion detection system through the use of an Autoencoder as a One-Class deep learning technique, with the objective of training a model with high learning capacity without the need for data labeling.*

Resumo. *A Internet das Coisas (IoT) é uma tendência global em evolução e está cada vez mais difundida. Apesar do seu potencial, a propagação de dispositivos IoT que possuem pouca segurança incorporada tem trazido grandes desafios para a segurança das redes. Portanto muitas propostas de sistemas de detecção de intrusão têm surgido a fim de mitigar este problema. No entanto, grande parte destes trabalhos utilizam-se de algoritmos supervisionados, o que torna-se um problema, pois a rotulação dos dados de treinamento é uma tarefa demorada e custosa. Como solução para este problema, este trabalho propõe um sistema de detecção de intrusão através da utilização de um Autoencoder como técnica de aprendizado profundo One-Class, com o objetivo de treinamento de um modelo com alta capacidade de aprendizado sem a necessidade de rotulação dos dados.*

1. Introdução

O crescimento do número de dispositivos de Internet das Coisas (*Internet of Things* - IoT) conectados à Internet trouxe diversos desafios para a segurança das redes. Apesar de gerar e manipular muitas informações privadas, esses dispositivos, normalmente devido ao baixo custo, possuem pouca ou nenhuma segurança incorporada [3].

Devido ao fato desses dispositivos não serem seguros como os computadores, mas ainda assim se envolverem em tarefas sensíveis à segurança e privacidade dos usuários, a IoT tem sido um alvo recorrente para pessoas mal intencionadas [4], que podem utilizar desses equipamentos para extrair informações confidenciais e também direcionar ataques de negação de serviço através de uma rede de *bots* em larga escala [14].

Com essas constantes ameaças, a criação de sistemas de detecção de intrusão (IDS) tem sido fundamental para esses dispositivos [2]. Estes sistemas são uma das principais ferramentas quando se trata de segurança das redes tradicionais. No entanto,

para dispositivos IoT ainda se mantém presentes alguns problemas em aberto devido às características particulares desta tecnologia, como recursos limitados, protocolos de comunicação específicos e comportamentos padronizados [6, 33].

Portanto, o desenvolvimento de sistemas de detecção de intrusão para IoT tem sido um grande desafio para pesquisadores de segurança da informação e administradores de redes. Pensando nisso, muitos trabalhos surgiram com o objetivo de mitigar essas ameaças. No entanto, muitos destes trabalhos se baseiam em técnicas de aprendizado supervisionado, que dificulta o treinamento do algoritmo, bem como a adaptação a novos ataques [10, 24, 25].

Por outro lado, algoritmos não supervisionados *One-Class* necessitam modelar apenas um único padrão e utilizar este para discernir se uma nova amostra pertence ou não ao padrão, ou seja, se é ou não um ataque. Isso leva a uma facilidade considerável de treinamento e retreinamento do algoritmo, uma vez que classificadores *Multi-Class* necessitam de amostras de todas as classes catalogadas no treinamento, algo que demanda um considerável esforço humano [3, 4, 27].

Pensando nisso, a proposta deste trabalho é criar um sistema de detecção de intrusão baseado em rede através de técnicas de aprendizado profundo *One-Class Classification*. A proposta é uma abordagem baseada em *deep learning*, que utiliza um autoencoder para realizar a detecção dos ataques.

Primeiramente é extraído um conjunto de características do tráfego normal dos dispositivos conectados na rede. Este conjunto de características será então utilizado para que o autoencoder aprenda o comportamento normal destes dispositivos. Portanto será utilizada uma rede neural para cada dispositivo conectado, uma vez que os comportamentos normais se diferem entre diferentes equipamentos. Quando novas amostras são adicionadas, o autoencoder tenta compactar as características, e quando ele falha em reconstruir as mesmas, é um forte indício de que a amostra observada é anômala.

A utilização de um autoencoder para essa abordagem traz diversas vantagens. Dentre elas, a sua excelente capacidade na aprendizagem de dados complexos se destaca, pois pode reduzir consideravelmente o número de falsos positivos quando comparado com outros algoritmos comumente utilizados com o mesmo objetivo [21]. Além disso, devido ao fato dos dispositivos IoT, normalmente, não possuírem uma vasta variedade de funções, utilizar o autoencoder para aprender o comportamento padrão certamente se torna uma tarefa simplificada, facilitando também a detecção de um ataque quando o comportamento se difere consideravelmente do padrão aprendido.

2. Fundamentação Teórico-Metodológica e Estado da Arte

2.1. Internet das Coisas

A Internet das Coisas (do inglês *Internet of Things*) pode ser descrita como uma rede de dispositivos físicos conectados entre si através de sensores, circuitos e softwares que tem como objetivo permitir a coleta e troca de dados entre eles [8].

Nos últimos anos, a Internet das Coisas vem em uma crescente considerável no cenário global da tecnologia. Estes equipamentos estão presentes em diversas áreas, como: transporte, infraestrutura civil, saúde, agricultura, indústrias, uso doméstico, etc [4].

Apesar de todas as melhorias e facilidades que esses equipamentos agregam a vida humana, uma única vulnerabilidade em tais sistemas pode levar a consequências que vão desde a perda de privacidade, danos físicos, financeiros e até mesmo a possibilidade de colocar vidas em risco [1]. Portanto, a segurança desses dispositivos tem sido um grande desafio para a comunidade científica, tendo em vista que, para esses equipamentos, a utilização de métodos convencionais de segurança são inadequados [33].

Isso ocorre devido a alguns fatos. Primeiramente, a arquitetura de rede desses dispositivos se difere das redes tradicionais. Em redes tradicionais, os sistemas finais se conectam diretamente com nós específicos (por exemplo, roteadores e pontos de acesso sem fio) que se responsabilizam por transmitir os pacotes ao destino. Por outro lado, em redes IoT, ao mesmo tempo que os dispositivos trabalham como sistemas finais, eles podem também ser responsáveis por realizar o encaminhamento dos pacotes, sem que haja a conexão com um nó específico para isso.

Além disso, esses dispositivos costumam usar protocolos de rede específicos, que não são utilizados em redes tradicionais, como o *IEEE 802.15.4*, *IPv6 over Low-power Wireless Personal Area Network (6LoWPAN)*, *IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL)* e *Constrained Application Protocol (CoAP)* [5, 29, 33]. Com o uso de diferentes protocolos, diferentes vulnerabilidades são apresentadas, dificultando a utilização de sistemas de detecção de intrusão convencionais.

Por fim, a segurança desses equipamentos se agrava devido a utilização de *hardwares* menos potentes, juntamente ao fato de que a maioria dos fabricantes não trabalham com questões relacionadas à segurança.

2.2. Sistemas de Detecção de Intrusão

Sistemas de Detecção de Intrusão (do inglês *Intrusion Detection System*) podem ser definidos como um *hardware*, ou um *software* ou uma combinação de ambos com o de objetivo monitorar uma rede a fim de encontrar eventos que possam violar suas regras de segurança [11, 13].

Com o grande aumento das ameaças pelas redes e a dificuldade que se tem para mitigação dessas ameaças, os Sistemas de Detecção de Intrusão têm recebido atenção de grande parte dos pesquisadores [17]. A principal função do sistema é detectar uma atividade suspeita e posteriormente relatar alertas aos administradores da rede em um tempo hábil.

Há alguns anos que os IDSs vêm sendo uma ferramenta de extrema importância na proteção, alerta e monitoramento de redes. Entretanto, a utilização de técnicas tradicionais associadas a dispositivos IoT não tem demonstrado bons resultados, isso ocorre devido ao fato desses dispositivos possuírem particularidades que os diferem de dispositivos convencionais, como recursos limitados, protocolos e arquitetura específicos [33].

Os IDSs podem ser classificados de duas diferentes formas: *Host-based Intrusion Detection Systems (HIDS)* e *Network-based Intrusion Detection Systems (NIDS)*. Os *Host-based IDSs* são instalados no próprio dispositivo sob monitoramento a fim de avaliar tanto os tráfegos na rede, como também recursos como uso de CPU, memória, energia, entre outros. Já os *Network-based NIDSs*, normalmente tem como objetivo monitorar apenas o tráfego de rede a fim de encontrar atividades maliciosas [12, 33].

Além disso, os IDSs podem ser implementados basicamente de duas diferentes formas. A primeira é conhecida como *signature-based IDS*. Neste caso, o sistema deve conhecer previamente os padrões utilizados em cada tipo de ataque que ele poderá identificar. A vantagem nesse caso é a rápida identificação quando um ataque conhecido ocorre. Sua desvantagem é justamente quando ocorre um ataque desconhecido, caso os padrões deste ataque não sejam similares a algum conhecido pelo sistema, poderá se passar como um tráfego normal [12, 23].

A segunda abordagem é conhecida como *anomaly-based IDS*. Neste caso, o sistema irá traçar os padrões e comportamentos normais no dispositivo ou tráfego de rede. Quando um comportamento que difere desses padrões ocorre, ele é catalogado como anômalo. A grande vantagem dessa abordagem é de que o sistema não necessita conhecer previamente os padrões dos ataques. A desvantagem é devido ao fato de normalmente gerar uma quantidade considerável de falsos positivos [4, 23].

2.3. Aprendizado de Máquina

Aprendizado de Máquina (do inglês *Machine Learning*) é uma subárea da Inteligência Artificial que tem como objetivo principal o desenvolvimento de técnicas computacionais com o intuito de encontrar funções aproximadas que possam tomar decisões com base em experiências acumuladas [22].

Existem basicamente dois tipos de problemas que são resolvidos por meio do aprendizado de máquina: regressão e classificação. Os problemas de regressão podem ser definidos resumidamente como uma predição de valores contínuos, por exemplo predição de salário, preço, idade, etc. Já os problemas de classificação baseiam-se em prever uma categoria de uma observação dada ao algoritmo. Por exemplo, dada uma amostra, dizer se ela é normal ou faz parte de um ataque.

Atualmente existem diversas categorias de aprendizado que os modelos de aprendizado de máquina podem se enquadrar. Uma delas é o aprendizado supervisionado. Neste caso o algoritmo depende de um conjunto de dados rotulados, ou seja, os dados utilizados para o treinamento devem conter as respostas desejadas. Durante o treinamento, o algoritmo recebe o conjunto de dados de entrada e para cada amostra ele faz uma predição, e posteriormente, com base no rótulo, a verficação do quão próximo ela foi do esperado. Desta forma, o algoritmo consegue fazer os ajustes necessários para que as predições sejam cada vez mais próximas aos rótulos.

No entanto, rotular os dados manualmente é uma tarefa demorada e custosa [18]. Para mitigar esse problema, existe a categoria dos algoritmos não supervisionados. Neste caso o objetivo é extrair informações valiosas sobre o conjunto de dados, descobrindo padrões ocultos ou agrupamentos de dados sem a necessidade de intervenção humana. Por exemplo, dado um tráfego de rede com diversos tipos de ataque, um algoritmo não supervisionado pode ser utilizado para dividir esse conjunto em diferentes categorias. Para isso ele irá buscar padrões e semelhanças entre os ataques, podendo fazer a distinção entre eles sem a necessidade de um rótulo para validação [1].

[1] `https://www.ibm.com/cloud/learn/machine-learning`

2.4. Redes Neurais

As Redes Neurais (do inglês *Neural Networks*) são uma subárea do aprendizado de máquina, que simula a maneira que as sinapses funcionam no cérebro humano. Enquanto as abordagens tradicionais de computação utilizam de séries de blocos para executar as tarefas, as redes neurais utilizam redes compostas por nós (simulando os neurônios) e arestas (simulando as sinapses) para processamento dos dados [31].

O primeiro modelo de rede neural foi proposto por Warren McCulloch e Walter Pitts em 1943 [20]. Através de um artigo, eles descreveram como os neurônios devem funcionar. Além disso, modelaram suas ideias por meio de uma rede neural simples com circuitos elétricos. Esse modelo foi precursor para o desenvolvimento das pesquisas na área.

As pesquisas aceleraram rapidamente, até que, em 1975, Kunihiko Fukushima apresentou pela primeira vez o conceito de rede neural multicamadas [7]. O conceito das redes neurais multicamadas se mantém até os dias atuais, e elas podem ser representadas conforme observado na Figura 1.

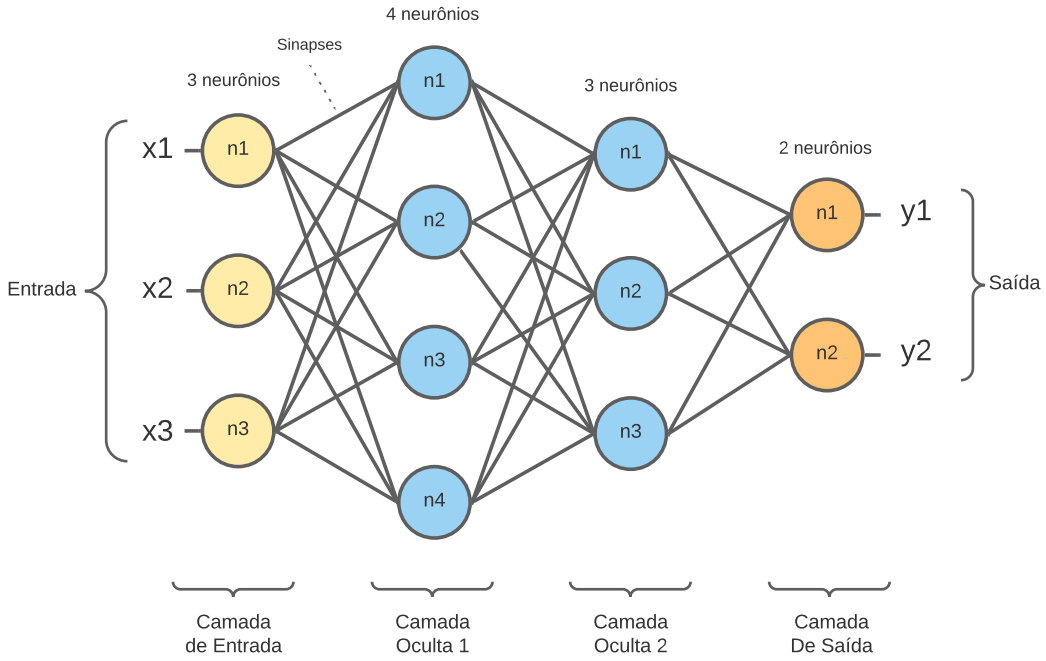

Figura 1. Rede neural multicamadas

Usualmente as camadas são classificadas em três diferentes grupos:

· **Camada de Entrada:** Onde é apresentado o conjunto de dados de entrada para a rede.
· **Camadas Ocultas:** Onde é realizada a maior parte do processamento; através das conexões ponderadas por pesos.
· **Camada de Saída:** Onde é apresentado o resultado final do processamento.

As redes neurais multicamadas são projetadas de forma que os neurônios sejam organizados em duas ou mais camadas de processamento, visto que sempre existirá ao

menos uma camada de entrada e uma de saída. Os neurônios se conectam por meio de arestas, e através destas conexões, os dados de entrada são processados pela rede até que se alcance a camada de saída.

2.5. Aprendizado Profundo

Aprendizado Profundo (do inglês *Deep Learning*) é uma subárea derivada das redes neurais, que é caracterizada por possuir uma quantidade considerável de neurônios quando comparados com outros tipos de redes neurais [28]. Portanto, o aprendizado profundo permite que modelos compostos de diversas camadas de processamento resultem em uma maior capacidade de abstração e generalização dos dados [16].

Devido a maior capacidade de processamento e abstração dos dados, esses algoritmos visam resolver problemas mais complexos, ao passo que necessitam de maior prática para alcançar os resultados.

3. Objetivos

O trabalho tem como objetivo principal o desenvolvimento e avaliação de um sistema de detecção de intrusão em redes IoT baseado em técnicas de aprendizado profundo. Para que este objetivo seja alcançado, os seguintes objetivos específicos foram definidos:

1. Estudar a fundamentação teórica, buscando o estado da arte por meio de revisões bibliográficas dos trabalhos relacionados com a utilização de técnicas de aprendizado profundo, bem como estudos que envolvam a detecção de ataques em dispositivos IoT.
2. Buscar conjuntos de dados que possuam os tráfegos de rede rotulados, para que seja possível realizar os testes e a validação da implementação que será realizada.
3. Selecionar as características que melhor evidenciam a ocorrência de ataques aos dispositivos IoT. Estas características serão um ponto crucial para que bons resultados sejam alcançados na detecção dos ataques.
4. Realizar a implementação dos algoritmos de pré-processamento e extração de características dos dados, bem como a implementação do Autoencoder que utilizará estas características para o treinamento.

4. Procedimentos metodológicos/Métodos e técnicas

Inicialmente, será realizada uma revisão bibliográfica a fim de encontrar as principais técnicas de aprendizado profundo, assim como trabalhos que utilizam dessas técnicas para detecção de ataques em redes.

Após a realização da leitura dos principais trabalhos envolvidos, serão escolhidas as características que melhor evidenciam a ocorrência de ataques em redes IoT. Estas devem ser cautelosamente selecionadas, pois a escolha destas características é importante para melhor precisão do modelo ao classificar novas amostras [19].

Com as características já selecionadas, serão realizadas buscas a fim de encontrar *datasets* contendo tráfegos de redes IoT. A implementação do algoritmo de aprendizado profundo se dará por meio de um Autoencoder, que é uma técnica de aprendizado de máquina, neste caso, classificada como *One-Class Classification (OCC)*, um caso especial de *Multi-Class Classification*, onde o treinamento é realizado por meio de uma única

classe positiva [27]. Portanto, serão necessários apenas tráfegos normais de redes para treinamento do algoritmo, que por sua vez facilitará na busca dos *datasets*.

Por fim, com as características e datasets escolhidos, será iniciada a implementação dos algoritmos de pré-processamento e extração das características dos *datasets* por meio de *scripts* implementados em Golang e Python, que posteriormente serão disponibilizados para a comunidade. Por último será implementado o Autoencoder, técnica de aprendizado profundo que será utilizada para o treinamento e classificação das amostras.

Após o treinamento, avaliação e comparação dos resultados obtidos, serão avaliados novos métodos com o intuito de aprimorar o Autoencoder e consequentemente melhorar os resultados. Serão estudadas técnicas de *Hyperparameter Optimization (HPO)*, como *Grid Search* e *Random Search*, com o intuito de encontrar os hiperparâmetros que obtenham os melhores resultados [32], bem como técnicas conhecidas como *Neural Architecture Search (NAS)*, que tem como finalidade encontrar a melhor arquitetura de rede neural, de acordo com o conjunto de dados de entrada [15].

Essas técnicas têm como objetivo minimizar o tempo e o esforço de trabalho humano em busca dos melhores resultados [9], visto que técnicas de aprendizado profundo possuem um grande número de variáveis; como: diversos hiperparâmetros, número de camadas, quantidade de neurônios por camada, etc. Além disso, futuramente poderão ser adicionadas técnicas de *Meta-Learning*, com o objetivo de encontrar não somente os melhores hiperparâmetros e estrutura da rede neural, como também os modelos de aprendizado profundo mais adequados para o conjunto de dados de entrada e os objetivos a serem alcançados [30].

5. Cronograma de Execução

Para que os objetivos sejam alcançados, as atividades serão divididas da seguinte maneira:

1. Levantamento bibliográfico.
2. Escolha das características que serão utilizadas para o treinamento e classificação dos conjuntos de dados.
3. Coleta dos *datasets* contendo tráfegos de redes IoT.
4. Implementação dos algoritmos de pré-processamento e extração das características mais relevantes.
5. Implementação e treinamento do Autoencoder.
6. Realização de testes e comparação dos resultados obtidos.
7. Estudo de técnicas como *Hyperparameter Optimization* e *Neural Architecture Search* com o intuito de obter melhores resultados.
8. Escrita do Trabalho de Conclusão de Curso.

Tabela 1. Cronograma de Execução

	set	out	nov	dez	jan	fev	mar	abr
Atividade 1	X	X						
Atividade 2		X						
Atividade 3		X						
Atividade 4			X	X				
Atividade 5			X	X	X			
Atividade 6					X	X		
Atividade 7						X	X	X
Atividade 8					X	X	X	X

6. Contribuições e/ou Resultados esperados

Dentre as principais contribuições esperadas, destaca-se a contribuição no aumento da segurança de redes IoT, sejam elas residenciais ou comerciais. Espera-se que as características selecionadas sejam relevantes o suficiente para que o Autoencoder possa diferenciar amostras normais de amostras de ataque para diversos tipos de ataques.

Além disso, espera-se melhores resultados com a utilização de aprendizado profundo quando comparado com métodos tradicionais para a detecção de ataques, tendo em vista que nos últimos anos, técnicas de aprendizado profundo têm demonstrado uma excelente capacidade na aprendizagem de dados complexos e temporais [26].

Por fim, que os *scripts* de pré-processamento de dados e extração de características facilitem e contribuam para o desenvolvimento de futuras pesquisas.

7. Espaço para assinaturas

Londrina, *13 de setembro de 2021*.

_____ _____

Vinicius Marino Luciano Bruno Bogaz Zarpelão

Referências

[1] Hezam Akram Abdul-Ghani, Dimitri Konstantas, and Mohammed Mahyoub. A comprehensive iot attacks survey based on a building-blocked reference model. *International Journal of Advanced Computer Science and Applications*, 9(3), 2018.

[2] Elisa Bertino and Nayeem Islam. Botnets and internet of things security. *Computer*, 50(2):76–79, 2017.

[3] Vitor Hugo Bezerra, Victor G Turrisi da Costa, Ricardo Augusto Martins, Sylvio Barbon Junior, Rodrigo Sanches Miani, and Bruno Bogaz Zarpelao. Providing iot host-based datasets for intrusion detection research. In *Anais do XVIII Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais*, pages 15–28. SBC, 2018.

[4] Vitor Hugo Bezerra, Victor Guilherme Turrisi da Costa, Sylvio Barbon Junior, Rodrigo Sanches Miani, and Bruno Bogaz Zarpelão. Iotds: A one-class classification approach to detect botnets in internet of things devices. *Sensors*, 19(14):3188, 2019.

[5] Walter Colitti, Kris Steenhaut, Niccolò De Caro, Bogdan Buta, and Virgil Dobrota. Evaluation of constrained application protocol for wireless sensor networks. In *2011 18th IEEE Workshop on Local & Metropolitan Area Networks (LANMAN)*, pages 1–6. IEEE, 2011.

[6] Mohamed Faisal Elrawy, Ali Ismail Awad, and Hesham FA Hamed. Intrusion detection systems for iot-based smart environments: a survey. *Journal of Cloud Computing*, 7(1):1–20, 2018.

[7] Kunihiko Fukushima. Cognitron: A self-organizing multilayered neural network. *Biological cybernetics*, 20(3):121–136, 1975.

[8] Pradyumna Gokhale, Omkar Bhat, and Sagar Bhat. Introduction to iot. *International Advanced Research Journal in Science, Engineering and Technology*, 5(1):41–44, 2018.

[9] Xin He, Kaiyong Zhao, and Xiaowen Chu. Automl: A survey of the state-of-the-art. *Knowledge-Based Systems*, 212:106622, 2021.

[10] Philokypros Ioulianou, Vasileios Vasilakis, Ioannis Moscholios, and Michael Logothetis. A signature-based intrusion detection system for the internet of things. *Information and Communication Technology Form*, 2018.

[11] Ja Jabez and B Muthukumar. Intrusion detection system (ids): Anomaly detection using outlier detection approach. *Procedia Computer Science*, 48:338–346, 2015.

[12] Ahmad Javaid, Quamar Niyaz, Weiqing Sun, and Mansoor Alam. A deep learning approach for network intrusion detection system. *Eai Endorsed Transactions on Security and Safety*, 3(9):e2, 2016.

[13] Shijoe Jose, D Malathi, Bharath Reddy, and Dorathi Jayaseeli. A survey on anomaly based host intrusion detection system. In *Journal of Physics: Conference Series*, page 012049. IOP Publishing, 2018.

[14] Constantinos Kolias, Georgios Kambourakis, Angelos Stavrou, and Jeffrey Voas. Ddos in the iot: Mirai and other botnets. *Computer*, 50(7):80–84, 2017.

[15] George Kyriakides and Konstantinos Margaritis. An introduction to neural architecture search for convolutional networks. *arXiv preprint arXiv:2005.11074*, 2020.

[16] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[17] Hung-Jen Liao, Chun-Hung Richard Lin, Ying-Chih Lin, and Kuang-Yuan Tung. Intrusion detection system: A comprehensive review. *Journal of Network and Computer Applications*, 36(1):16–24, 2013.

[18] Hongyu Liu and Bo Lang. Machine learning and deep learning methods for intrusion detection systems: A survey. *Applied Sciences*, 9(20), 2019.

[19] Huiqing Liu, Jinyan Li, and Limsoon Wong. A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns.

Genome informatics. International Conference on Genome Informatics, 13:51–60, 01 2002.

[20] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[21] Yair Meidan, Michael Bohadana, Yael Mathov, Yisroel Mirsky, Asaf Shabtai, Dominik Breitenbacher, and Yuval Elovici. N-baiot—network-based detection of iot botnet attacks using deep autoencoders. *IEEE Pervasive Computing*, 17(3):12–22, 2018.

[22] Maria Carolina Monard and José Augusto Baranauskas. Conceitos sobre aprendizado de máquina. *Sistemas inteligentes-Fundamentos e aplicações*, 1(1):32, 2003.

[23] Matheus P Novaes, Luiz F Carvalho, Jaime Lloret, and Mario Lemes Proença Jr. Adversarial deep learning approach detection and defense against ddos attacks in sdn environments. *Future Generation Computer Systems*, 125:156–167, 2021.

[24] Yazan Otoum, Dandan Liu, and Amiya Nayak. Dl-ids: a deep learning–based intrusion detection framework for securing iot. *Transactions on Emerging Telecommunications Technologies*, page e3803, 2019.

[25] Mete Ozay, Inaki Esnaola, Fatos Tunay Yarman Vural, Sanjeev R Kulkarni, and H Vincent Poor. Machine learning methods for attack detection in the smart grid. *IEEE transactions on neural networks and learning systems*, 27(8):1773–1786, 2015.

[26] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection. *ACM Computing Surveys*, 54(2):1–38, Apr 2021.

[27] Pramuditha Perera, Poojan Oza, and Vishal M Patel. One-class classification: A survey. *arXiv preprint arXiv:2101.03064*, 2021.

[28] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[29] Matthew Sherburne, Randy Marchany, and Joseph Tront. Implementing moving target ipv6 defense to secure 6lowpan in the internet of things and smart grid. In *Proceedings of the 9th Annual Cyber and Information Security Research Conference*, pages 37–40, 2014.

[30] Joaquin Vanschoren. Meta-learning: A survey. *arXiv preprint arXiv:1810.03548*, 2018.

[31] Sun-Chong Wang. Artificial neural network. In *Interdisciplinary computing in java programming*, pages 81–100. Springer, 2003.

[32] Tong Yu and Hong Zhu. Hyper-parameter optimization: A review of algorithms and applications. *arXiv preprint arXiv:2003.05689*, 2020.

[33] Bruno Bogaz Zarpelão, Rodrigo Sanches Miani, Cláudio Toshio Kawakani, and Sean Carlisto de Alvarenga. A survey of intrusion detection in internet of things. *Journal of Network and Computer Applications*, 84:25–37, 2017.