UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

João Wojtyla Ferreira de Mendonça

Robustez adversarial em sistemas de detecção de intrusão para redes: treino adversarial e detecção out-of-distribution

Recife

2025

João Wojtyla Ferreira de Mendonça

Robustez adversarial em sistemas de detecção de intrusão para redes: treino adversarial e detecção out-of-distribution

Dissertação de mestrado apresentada ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Federal de Pernambuco, como requisito parcial para obtenção do título de Mestre em Ciência da Computação

Área de Concentração: Inteligência computacional

Orientador (a): Cleber Zanchettin

Coorientador (a): Luís Fred Gonçalves de Sousa

Recife

2025

.Catalogação de Publicação na Fonte. UFPE - Biblioteca Central

Mendonca, Joao Wojtyla Ferreira de.
Robustez adversarial em sistemas de detecção de intrusão para redes: treino adversarial e detecção out-of-distribution / Joao Wojtyla Ferreira de Mendonca. - Recife, 2025.
110f.: il.

Dissertação (Mestrado) - Universidade Federal de Pernambuco, Centro de Informática, Programa de Pós Graduação Acadêmico em Ciências da Computação, 2025.
Orientação: Cleber Zanchettin.
Coorientação: Luís Fred Gonçalves de Sousa.
Inclui referências e anexos.

1. Redes neurais profundas; 2. Robustez adversarial; 3. Fora de distribuição. I. Zanchettin, Cleber. II. Sousa, Luís Fred Gonçalves de. III. Título.

UFPE-Biblioteca Central

João Wojtyla Ferreira de Mendonça

“Robustez adversarial em sistemas de detecção de intrusão para redes: treino adversarial e detecção out-of-distribution”

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovado em: 29/07/2025.

BANCA EXAMINADORA

_____

Prof. Dr. Divanilson Rodrigo de Sousa Campelo
Centro de Informática / UFPE

_____

Prof. Dr. João Fausto Lorenzato de Oliveira
Departamento de Engenharia de Controle
e Automação/UPE

_____

Prof. Dr. Prof. Dr Cleber Zanchettin
Centro de Informática / UFPE
(orientador)

AGRADECIMENTOS

Gostaria primeiramente de agradecer a meus Pais e familiares: Alberes Mendonça e Jadiel Mendonça pelo suporte emocional ao longo deste ciclo.

Aos amigos: Sônia Agostinho,José Alves Filho, Amanda Araújo, Klerysson Garcia, Leilane Cruz, José Wilson Barbosa e Paulo Costa. A vocês muito obrigado por terem me ajudado a chegar até aqui. Gostaria de incluir os integrantes do projeto MAIDAI: Prof. Cleber Zanchettin, Prof. Divanilson Campelo, Prof. Paulo Freitas, Luis Fred e Thiago Bispo. Cada conversa e sugestão nas reuniões foi de suma importância para a realização deste trabalho.

Meu muito obrigado a todos vocês.

RESUMO

Sistemas de detecção de intrusão baseados em aprendizado profundo são vulneráveis a ações evasivas chamadas ataques adversários. Este é um problema crítico, pois sistemas com suposta alta precisão são suscetíveis a ataques que não são identificados. Neste trabalho, apresentamos uma nova abordagem chamada TabIDS que utiliza treinamento adversário para aumentar a robustez classificatória contra ataques adversariais capazes de evadir sistemas de detecção de intrusão, sendo também capaz de detectar ataques não vistos no treinamento do modelo. Para isso, foram implementados dois classificadores para compor um sistema de detecção de intrusão em redes (Network Intrusion Detection System (NIDS)): um para detecção de ataques de rede e outro para amostras *out-of-distribution* Out-of-distribution (OOD). O método desenvolvido combina o modelo treinado adversarialmente com um método por distância Mahalanobis projetado para detectar amostras adversárias como sendo OOD. Dessa forma é preenchida uma lacuna onde ou o classificador apenas consegue lidar com um tipo de ataque adversarial, ou o detector OOD é aplicado apenas para amostras não perturbadas adversarialmente. Foi avaliado o desempenho do modelo nos conjuntos de dados CIC IDS2017 e UNSW-NB15, e a abordagem utilizando treinamento adversário com detecção de OOD se mostrou robusta contra ataques adversários avançados e mesmo contra ataques de rede não vistos no treinamento. Os resultados obtidos demonstram que o TabIDS supera modelos convencionais em Precisão, Recall e Área sob a curva ROC (ROC-AUC), especialmente em cenários adversariais com perturbações imperceptíveis. A detecção de OOD baseada em distância de Mahalanobis atingiu até 99,5% de AUC em alguns ataques, destacando a eficácia do método proposto. Os resultados mostram que a abordagem é promissora para aplicações de cibersegurança que demandam robustez e generalização frente a ataques desconhecidos.

Palavras-chaves: Dados fora de distribuição. Robustez adversarial. Sistemas de detecção de intrusão.

ABSTRACT

Deep learning-based intrusion detection systems are vulnerable to evasive actions called adversarial attacks. This is a critical problem, as systems with supposedly high accuracy are susceptible to attacks that are not identified. In this work, we present a new approach called TabIDS that uses adversarial training to increase classification robustness against adversarial attacks capable of evading intrusion detection systems, while also being able to detect attacks not seen during model training. To this end, two classifiers were implemented to compose a network intrusion detection system (NIDS): one for detecting network attacks and the other for OOD samples. The developed method combines the adversarially trained model with a Mahalanobis distance method designed to detect adversarial samples as OOD. This fills a gap where either the classifier can only handle one type of adversarial attack, or the OOD detector is applied only to samples not adversarially perturbed. The model's performance was evaluated on the CIC IDS2017 and UNSW-NB15 datasets, and the approach, using adversarial training with OOD detection, proved robust against advanced adversarial attacks and even against network attacks not seen in the training. The results demonstrate that TabIDS outperforms conventional models in Precision, Recall, and ROC-AUC, especially in adversarial scenarios with imperceptible perturbations. OOD detection based on Mahalanobis distance achieved up to 99.5% AUC in some attacks, highlighting the effectiveness of the proposed method. The results show that the approach is promising for cybersecurity applications that require robustness and generalization against unknown attacks.

Keywords: Out-of-distribution, Adversarial Robustness, Network Intrusion Detection System.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AdvML Adversarial Machine Learning

APT Advanced Persistent Threats

AT-MPB Adversarial Training with Multiple Perturbation Bounds

AUC Area Under Curve

BIM Basic Iterative Method

CAN Controller Area Network

CW Carlini-Wagner

DA Domain Adaptation

DDoS Distributed Denial of Service

DG Domain Generalization

DL Deep Learning

DoS Denial of Service

FGSM Fast Gradient Signed Method

FP False Positive

FPR False Positive Rate

GAN Generative Adversarial Netork

HIDS Host Intrusion Detection System

ID In-distribution

IDS Intrusion Detection System

IoT Internet of Things

IPS Intrusion Prevention System

IRM Invariant Risk Minimization

KNN K-Nearest Neighbors

LSTM Long Short-Term Memory

MITRE ATT&CK MITRE - Adversarial Tactics, Techniques and Common Knowledge

ML Machine Learning

MLP Multi-Layer Perceptron

NIDS Network Intrusion Detection System

OOD Out-of-distribution

PGD Projected Gradient Descent

PN Prior Networks

ROC Receiver Operation Characteristic

SDN Sofware-Defined Networking

TP True Positive

TPR True Positive Rate

LISTA DE SÍMBOLOS

α	Taxa de aprendizado de um ataque adversarial baseado em gradiente
δ	Perturbação adversarial
$\mathcal{D}$:	Distribuição dos dados de treinamento.
f	Função que representa um modelo de rede neural
$\mathcal{L}$	Função de perda
$\|a - b\|$	Valor absoluto da diferença entre dois valores quaisquer
$\|\|\ \|\|$	Norma de um vetor
ℓ_2	Norma calculada por: $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$
L_2	Métrica de distância a partir norma ℓ_2
ℓ_∞	Norma calculada por: $\|\mathbf{x}\|_\infty = \max_i \|x_i\|$
L_∞	Métrica de distância a partir da norma ℓ_∞
$\mathbf{m}$	Máscara binária
$\odot$	Produto elemento a elemento.
$\mathcal{S}$:	Conjunto de restrições $\mathcal{S}$ (como uma bola ℓ_p).
$\sum_{i=1}^{n}$	Somatório a partir de i até n
$\int_a^b$	Integral definida entre a e b
$\mathbf{x}$	Amostras sem perturbação adversarial
$\mathbf{x}_{adv}$	Amostras perturbadas por ataque adversarial
y	Rótulo dos dados usados para classificação.
$\nabla_\delta \mathcal{L}$	Gradiente da função de perda em relação a δ
θ	Parâmetros do modelo.

SUMÁRIO

1 INTRODUÇÃO

O número de dispositivos conectados em rede tem crescido exponencialmente. A internet, ao proporcionar uma comunicação ágil, tornou as redes de computadores onipresentes no ambiente corporativo. Uma consequência direta deste cenário é o aumento anual dos ciberataques tendo estas redes como alvo, o que exige investimentos significativamente elevados para prevenir ou mitigar os efeitos adversos causados por esses ataques Fig. 1.

Ataques a redes de computadores estão evoluindo para formas de ataque multi estágio e de longo prazo. Como exemplo: os ataques DDoS, que dependem da propagação de botnets em computadores vulneráveis, para depois causarem a negação de serviço ao sistema alvo; e as Advanced Persistent Threats (APT), que são capazes de permanecer furtivas na rede do alvo por longos períodos de tempo, conforme a necessidade do invasor, resultando em prejuízos substanciais tais como perdas financeiras elevadas ou vazamentos de dados sensíveis Kim, Wang e Ullrich 2012, Kaspersky 2022.

A intrusão via redes Uma e Padmavathi 2013 é das mais sérias ameaças a sistemas corporativos de computadores, uma vez que a partir dela o atacante pode causar danos irrecuperáveis para as organizações, incluindo vazamento de dados financeiros, dados de fornecedores, e interrupção no fornecimento de serviços críticos Jeba et al. 2024. Durante essas invasões, o adversário geralmente consegue instalar malwares que facilitem o seu acesso a outras redes, omitindo sua localização e identificação da rede/computador de origem Jiang, Wu e Xin 2022. Em razão disso, faz-se mandatório o desenvolvimento de novas técnicas capazes de evitar ou detectar o acesso malicioso. Isto inclui os sistemas de detecção de intrusão Abdulganiyu, Tchakoucht e Saheed 2023.

De acordo com Masdari e Khezri 2020 a detecção de ameaças em um Intrusion Detection System (IDS) tem sido baseada num conjunto de técnicas que incluem a detecção de anomalias, detecção de uso malicioso com base em assinaturas de ataques conhecidos e métodos híbridos. No IDS baseado em detecção de anomalias, o perfil de comportamento normal deve ser definido antecipadamente. Qualquer desvio significativo em relação a esta norma pode ser considerado uma anomalia comportamental característica de uma invasão Masdari e Khezri 2020.Embora os IDSs baseados em anomalias possam lidar com novos tipos de ataques, definir e atualizar o comportamento normal pode ser um desafio em organizações grandes e dinâmicas Farahnakian e Heikkonen 2018.

Por outro lado, em You et al. 2022 é dito que:

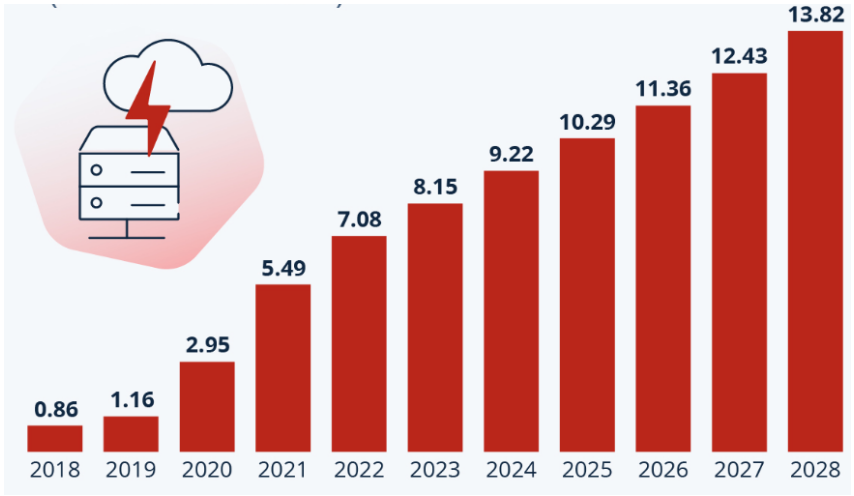
- A detecção baseada em assinaturas, que compara o padrão de uma ameaça em curso com regras pré-existentes, não é eficaz na detecção de novos tipos de ataques, muito embora apresentem alta precisão de detecção para os ataques conhecidos.
- Além disso, elas experimentam muitos casos de falsos positivos para ameaças inéditas, algo que limita suas implantações em sistemas críticos.

Os desafios encontrados ao detectar anomalias em cibersegurança :

- Anomalias geralmente são raras, levando a dificuldades na modelagem e treinamento de algoritmos. Isso dificulta a construção de modelos supervisionados e pode levar a vieses em favor da classe majoritária, gerando altas taxas de falsos negativos ou falsos positivos Kohli e Chhabra 2025.
- Problemas de qualidade dos dados, como ruído, entradas ausentes ou inconsistentes, podem degradar drasticamente o desempenho. Modelos avançados requerem pre-processamento robusto para lidar com esses problemas. Algoritmos modernos, redes neurais profundas, têm poder para capturar essas complexidades, mas exigem recursos computacionais e ajustamentos sofisticados Kohli e Chhabra 2025.
- Robustez a ruídos e variações contextuais Dados reais são ruidosos e podem conter outliers não representativos de anomalias verdadeiras. Além disso, anomalias podem ser contextuais (válidas em um contexto, anômalas em outro), exigindo modelagens sensíveis a variáveis externas e variações de escala Adhikari et al. 2024.

Para abordar essas questões, técnicas convencionais de aprendizado de máquina têm sido amplamente utilizadas para detecção de intrusão. Os trabalhos Yang et al. 2022 e Elsayed, Mohamed e Madkour 2024 mencionam que os modelos tradicionais de aprendizado de máquina, tais como árvores de decisão, máquinas de vetor de suporte e random forest falham em detectar os ataques em dados de alta volumetria, são menos robustos a ruídos e outliers e têm dificuldade de extrair padrões de dados complexos. Ainda em Yang et al. 2022 os algoritmos tradicionais de aprendizagem de máquina geralmente sofrem com a alta dimensionalidade, uma solução comum é usar técnicas de pré-processamento de dados que podem ajudar a reduzir a dimensionalidade, contudo os métodos de pré-processamento (ex, redução de dimensionalidade e seleção de features) podem afetar o

Figura 1. Prejuízo causado por ciberataques ao longos dos anos. A partir de 2024 são projeções de gastos.



Fonte: https://www.weforum.org/stories/2024/07/crowdstrike-global-it-outage-cybersecurity-news-july-2024/.Acessado em 14-02-2025.

desempenho da detecção ao acrescentar uma camada de latência no pipeline, logo devem ser cuidadosamente considerados no projeto de métodos de detecção de intrusão, enquanto essas mesmas técnicas de pré-processamento não são imprescindíveis em Deep Learning (DL). Modelos de DL são úteis para ambientes de rede complexos e de grande escala, com o potencial para extrair padrões distintos dos dados sem que para isso seja necessário recorrer a técnicas de *feature engineering* Kouliaridis, Kambourakis e Geneiatakis 2020. Como resultado, vários pesquisadores neste campo se concentram no desenvolvimento de IDS baseados em DL, Farhan et al. 2025,Zhang et al. 2022,Liao et al. 2024,Elsayed, Mohamed e Madkour 2024, uma vez que conseguem detectar ameaças via processo de detecção de anomalias nos dados, ao invés de depender de assinaturas de ameaças conhecidas Yi et al. 2022.

Não obstante, modelos DL são vulneráveis a ataques adversariais, os quais são constituídos por pequenas perturbações adicionadas ao modelo com o objetivo de modificar a saída Goodfellow, Shlens e Szegedy 2015. No presente trabalho a robustez de um modelo é entendida como sua capacidade manter a performance classificatória em presença das amostras adversariais Zhang et al. 2019,Tsipras et al. 2019, Madry et al. 2017. Tal avaliação pode evitar que agentes maliciosos explorem a vulnerabilidade dos modelos e evadam o IDS Jmila e Khedher 2022 causando danos financeiros e a reputação da empresa atingida.

Em particular, Lin, Shi e Xue 2022, Shu et al. 2020 e Wu et al. 2019 demonstram a aplicabilidade de ataques adversariais evasivos a dados estruturados, introduzindo uma

vulnerabilidade significativa a aplicações de segurança cibernética.

Além disso, é pertinente enfatizar que as redes neurais profundas podem degradar significativamente o seu desempenho preditivo em razão de mudanças na distribuição dos dados. Isso ocorre porque uma premissa dos modelos é que os dados de teste mantenham a distribuição dos dados de treino; no entanto, isso não é factível ao colocar o modelo em produção Bulusu et al. 2020. Portanto, é evidente que um modelo confiável deve conseguir não apenas reconhecer amostras In-distribution (ID) como também OOD, desse modo evita-se que o modelo atribua erroneamente um alto nível de confiança a classes desconhecidas ou a amostras adversariais.

Uma solução direta para contornar o problema de dados OOD é coletar alguns dados do domínio de destino para adaptar um modelo treinado no domínio de origem Zhou et al. 2021. Esse problema de adaptação de domínio – usualmente mencionado como Domain Adaptation (DA) na literatura relevante – vem recebendo crescente atenção Lu et al. 2020, Saito et al. 2017 e Long et al. 2015 ao longo dos anos. No entanto, a DA depende de uma forte suposição de que os dados de destino são acessíveis para adaptação do modelo, o que nem sempre se mantém na prática e faz DA ficar alinhada a um domínio em particular. Na verdade, em muitas aplicações, os dados de destino são difíceis de obter ou mesmo desconhecidos antes de implementar o modelo.

Por exemplo, na segmentação semântica de cenas de tráfego, é inviável coletar dados capturando todas as cenas diferentes e sob todas as condições climáticas possíveis Yue et al. 2019. Para superar o problema de mudança de domínio, bem como a ausência de dados de destino, a abordagem generalização de domínio (Domain Generalization (DG)) foi introduzida Blanchard, Lee e Scott 2011. Especificamente, o objetivo em DG é otimizar um modelo usando dados de um ou múltiplos domínios de origem relacionados, mas distintos entre si, de tal forma que o modelo possa generalizar bem para qualquer domínio de destino a partir do aprendizado de representações invariantes. Para dar um exemplo, pode-se treinar um modelo meteorológico usando imagens diurnas com neve ou chuva e testar em imagens com neblina, que por não serem vistas no treinamento será detectada como OOD ao invés de o modelo errar com alta confiança.

Vários métodos lidam com o problema de generalização de OOD. Isto inclui métodos baseados no alinhamento de distribuições de domínio de origem para DG Li et al. 2018. Métodos que expõem o modelo à mudança de domínio durante o treinamento via meta-aprendizagem Balaji, Sankaranarayanan e Chellappa 2018 e métodos que aumentam dados

com síntese de domínio Zhou et al. 2020. Além disso, há também os modelos robustos a amostras adversariais, como apontados por Lee et al. 2018 e Malinin e Gales 2019.

Deve ser dito que nos trabalhos anteriores em detecção de intrusão para redes, tais como o trabalho de Hashemi e Keller 2020, que apesar de levar em consideração as restrições em dados tabulares, os autores usam ataques adversariais white box sem adaptação apropriada aos dados de fluxo de rede, potencialmente gerando amostras inválidas. Liang et al. 2022 embora não utilizou as restrições de dados tabulares, porém utiliza um detector OOD para amostras não vistas no treinamento. Ceccarelli e Zoppi 2023 utilizaram ataques black box seguindo as restrições e uso de out of distribution para amostras clean, portanto é um método que pode não ser seguro a ataques white-box ou mesmo a ataques que evadam o detector OOD. Paya et al. 2023 usaram ataques white box baseado em um modelo de ameaça *Insider Threat* e com restrições para dados tabulares, no entanto o método dos autores detecta ataques black-box, mas não é robusto a amostras adversariais white-box. O presente trabalho traz como diferencial: utilizar ataques white e black box modificados para que as amostras sejam estimadas pelos métodos a partir da restrição por máscara binária. E tanto ataques black box baseados em consultas como os de transferência. O detector OOD é usado para detectar classes não vistas no treinamento e amostras adversariais.

1.1 OBJETIVO

No presente trabalho é proposta uma metodologia em duas etapas para robustez adversarial e detecção de dados OOD em um NIDS. Na primeira etapa, é proposto um classificador treinado adversarialmente e na segunda etapa, um detector os dados OOD.

1.2 OBJETIVOS ESPECÍFICOS

- Desenvolver um modelo deep learning para segurança de rede, porém robusto contra ataques adversariais;
- Implementar um esquema em duas etapas, tornar o sistema de detecção de intrusão capaz de detectar amostras OOD;
- Fazer com que o modelo proposto seja tanto robusto a amostras adversariais quanto

capaz de detectá-las, deste modo tornando-o apropriado para detecção de uma maior diversidade de ataques;

1.3 APRESENTAÇÃO

Este trabalho está dividido com a seguinte organização: o Capítulo 1 apresenta o problema e as motivações que levaram a esta pesquisa. O Capítulo 2 apresenta uma explicação sobre detecção de ataques em redes e demonstra os principais tipos de implementação para detecção. Também são expostos alguns conceitos básicos para o entendimento deste trabalho e apresentar uma revisão bibliográfica para os conceitos apresentados, e por fim apresentar o estado presente dos artigos relacionados ao tema. No Capítulo 3 é apresentada a proposta deste trabalho. No Capítulo 4 são apresentados os experimentos e resultados. O Capítulo 5 apresenta as conclusões e sugestões para trabalhos futuros.

2 DETECÇÃO DE INTRUSÃO

2.1 ATAQUES A REDES

Redes de computadores, sejam públicas ou privadas, são vulneráveis a diversas formas de invasão. Por exemplo, um usuário pode baixar arquivos de um site, e um desses arquivos pode roubar e enviar informações privadas pela internet. Outra tática comum é o invasor se passar por uma pessoa de confiança para obter dados confidenciais. Em ambos os cenários, a interação do usuário é um fator crítico para o sucesso da invasão Bishop 2018.

Todavia, há ataques que exploram vulnerabilidades a partir do contato com os sistemas operacionais presentes nos dispositivos conectados. Esses ataques são realizados diretamente via rede e parecem fluxos de tráfego de rede normais. Uma intrusão de rede pode ser passiva (caracterizada pela obtenção de acesso de forma silenciosa e indetectável, com o objetivo principal de coletar informações) ou ativa (envolvendo a obtenção de acesso secreto para realizar modificações nos recursos da rede, como a alteração de dados ou configurações) Bishop 2018.

Nos ataques passivos, o invasor apenas monitora a rede da vítima, analisando o fluxo de tráfego ou verificando portas abertas. Isso ajuda o adversário a obter informações sobre quais portas são muito usadas e quais estão ociosas. O principal objetivo dos ataques passivos é coletar informações sobre o sistema alvo. Eles não pretendem prejudicar o funcionamento normal dos sistemas, porque querem passar despercebidos enquanto roubam informações. Ataques passivos podem ocasionalmente abrir o caminho para ataques ativos na rede alvo Stallings e Brown 2018.

Durante uma intrusão do tipo *ativa*, o invasor interfere ativamente no fluxo de dados em uma rede e no funcionamento de um determinado dispositivo, instalando malwares capazes de interromper o funcionamento do sistema ou vazar informações importantes para o atacante. Ataques do tipo Negação de Serviço Denial of Service (DoS) são exemplos de intrusão ativa. Neste ataque, o invasor ocupa a largura de banda do sistema alvo e o mantém ocupado para que ele não possa atender solicitações de outras máquinas internas ou externas. Uma variação do DoS é o ataque de Negação de Serviço Distribuído Distributed Denial of Service (DDoS), onde vários invasores têm como alvo o mesmo sistema de diferentes endereços IP e diferentes locais Bishop 2018.

Existem também ataques que visam coletar dados confidenciais de uma só vez ou

como parte de uma conexão parasitária de longo prazo que continuará a sugar dados até que sejam identificados. Alguns intrusos tentarão implantar um código que irá quebrar senhas, capturar ações de pressão de teclas ou clonar o site enquanto redireciona usuários desavisados para seu próprio endereço malicioso. Outros irão se infiltrar na rede, desviando furtivamente dados regularmente ou alterando páginas Web públicas com mensagens variadas Vacca 2013.

Inúmeras técnicas foram desenvolvidas e aprimoradas ao longo do tempo com o objetivo de prevenir que redes de computadores sejam alvos de ataques maliciosos. Dentre essas abordagens, uma das mais difundidas e eficazes é o IDS. O IDS atua como uma ferramenta de segurança proativa, monitorando continuamente o tráfego de rede e os eventos dos sistemas em busca de padrões ou atividades que possam indicar uma tentativa de intrusão Yin et al. 2023. Ao identificar comportamentos suspeitos, o IDS gera alertas, permitindo que os administradores de rede tomem medidas corretivas em tempo hábil, como bloquear o tráfego de origem do ataque ou isolar sistemas comprometidos.

Para auxiliar na modelagem de ameaças cibernéticas um framework muito importante é o MITRE - Adversarial Tactics, Techniques and Common Knowledge (MITRE ATT&CK).

2.2 MITRE ATT&CK

O MITRE ATT&CK MITRE Corporation 2015-2025 é um framework largamente aplicado em indústrias associadas a saúde, finanças e infraestrutura crítica Li, Huang e Chen 2024. De acordo com Strom et al. 2020 é uma base de conhecimento organizada para descrever métodos empregados em ciberataques. Propõe uma taxonomia para descrever o comportamento dos invasores ao longo do ciclo de vida de um ataque. Abrange uma ampla gama de sistemas e estratégias, sendo amplamente utilizados no compartilhamento de dados sobre ameaças tecnológicas. Com o MITRE ATT&CK, um sistema de classificação está em vigor para vários comportamentos hostis, dividido em três categorias:

- Empresarial: Detalha comportamentos em sistemas de TI típicos, como Linux ou Windows.
- Mobile: Direcionado a dispositivos móveis, por exemplo, Android e iOS.

- ICS (*Industrial Control Systems*): Voltado para reguladores industriais e, em maior extensão, a sistemas ciberfísicos.

Sua estrutura se baseia nos seguintes conceitos-chave:

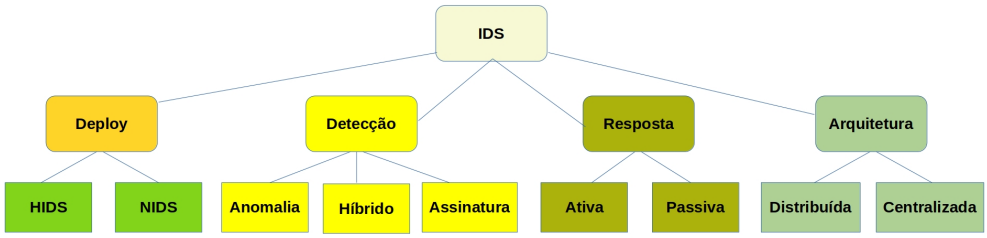
- Tática: Refere-se ao que um invasor faz para atingir um objetivo (descoberta, acesso inicial, persistência e etc.).
- Técnica: É uma maneira de realizar uma atividade, representando como um intruso atinge um objetivo tático por meio de ações.
- Procedimento: Refere-se a uma série de etapas bem definidas, uma instância particular do uso de uma técnica específica e descreve como um adversário implementa essa técnica. Dentro deste conceito as técnicas são vistas como ações individuais ou discretas e as táticas são a maneira de combinar essas ações.

No trabalho de Marinho e Holanda 2023 encontra-se o seguinte exemplo: suponha que um agente malicioso pretende roubar dados confidenciais armazenados em um servidor de uma empresa. Para fazê-lo, o intruso precisa encontrar uma maneira de entrar no sistema alvo, em seguida se mover de host em host até chegar ao servidor desejado, finalmente coletar e roubar os dados. A partir deste exemplo, entrar na rede alvo seria uma *Tática* do tipo "acesso inicial"e pode ser realizada pela *Técnica* "credenciais válidas". Portanto, cada um dos movimentos necessários, desde entrar na rede empresarial até o roubo dos dados, pode ser mapeado na base MITRE ATT&CK, como também Táticas, Técnicas e Procedimentos para mitigação de ataques cibernéticos.

2.3 SISTEMAS DE DETECÇÃO DE INTRUSÃO

Uma intrusão em um sistema ou rede é uma tentativa intencional não autorizada, com ou sem êxito de: acessar, manipular, destruir ou usar indevidamente algum recurso computacional e onde o uso indevido pode resultar ou tornar a propriedade não confiável ou inutilizável Kizza 2024, Yin et al. 2020. Com o aumento da dependência das pessoas em relação à tecnologia, disparou uma nova onda de crimes relacionados a computadores.

Figura 3 – Classificação dos Sistemas de detecção de Intrusão.



2.3.1 Detecção de intrusão pelo tipo de implamentação

A Figura 3 apresenta uma classificação dos tipos de IDS. Dependendo do tipo de implementação, ou *deploy* , existem dois tipos de IDSs: o Sistema de Detecção de Intrusão de Rede (NIDS) e o Sistema de Detecção de Intrusão de Host (Host Intrusion Detection System (HIDS)). Nos sistemas NIDS, o tráfego malicioso é detectado utilizando todos os metadados e conteúdos de pacotes na rede. Em contraste, os sistemas HIDS realizam a detecção de intrusão em apenas um endpoint, sendo capazes de proteger contra ameaças internas e externas. Além disso, os sistemas HIDS formam uma camada de proteção adicional, já que possuem a vantagem de conseguir detectar ataques que podem não ser identificados pelo NIDS Mahdavifar e Ghorbani 2019.

2.3.2 Detecção de intrusão pelo tipo de detecção

Conforme o método de detecão, os modelos IDS podem ser entendidos como baseados em Assinatura ou baseados em Anomalia. A detecção baseada em assinatura funciona melhor para identificar ameaças conhecidas, onde detecta tráfego malicioso com base em regras predefinidas Farahnakian e Heikkonen 2018. O IDS baseado em detecção de anomalia detecta comportamento anormal ao modelar o comportamento normal por meio da extração de padrões. Normalmente, o IDS baseado em anomalia pode descobrir ataques complexos e desconhecidos, portanto, tendo melhor desempenho do que o IDS baseado em assinatura para ataques novos. Existe também o tipo híbrido, o qual combina os baseados em assinatura com os baseados em anomalias Bishop 2018. Desse modo há uma diminuição de falsos positivos e a possibilidade de detecção de ataques que não estão presentes no banco de dados do módulo baseado em assinatura. Entretanto são mais difíceis de configurar, o que pode gerar uma classificação discordante entre os detectores

e mais caros de implementar e manter Kizza 2024.

2.3.3 Detecção de intrusão pelo tipo de resposta

Conforme discutido em Thakkar e Lohiya 2020, um IDS pode ser classificado como ativo ou passivo, com base em seu mecanismo de resposta quando um ataque é detectado. Um IDS ativo, também conhecido como Sistema de Prevenção de Intrusão (Intrusion Prevention System (IPS)), é configurado de tal forma que, assim que um ataque é detectado, o sistema bloqueia automaticamente esses ataques sem nem mesmo consultar o analista de segurança. Esse IDS fornece uma resposta em tempo real disparando um alarme quando o ataque é detectado, bloqueando o ataque, gerando um relatório, criando um backup e registrando todas as informações. Um IDS passivo, por outro lado, é configurado para escanear e analisar o tráfego de rede e alertar o analista de rede para tomar outras medidas, como bloquear endereços IP, encerrar a conexão ou processo e bloquear a conta do usuário. Thakkar e Lohiya 2020 ainda argumentam que um IDS passivo é mais fácil de configurar e instalar e é menos suscetível a ataques em comparação com um IDS ativo.

2.3.4 Detecção de intrusão pelo tipo de arquitetura

Com base nos requisitos de infraestrutura, um IDS pode ser classificado como centralizado e distribuído Thakkar e Lohiya 2020. Um IDS centralizado fica instalado em um dispositivo central que é responsável por analisar o tráfego de rede e gerar um alarme se algum padrão anormal for detectado. Essas informações são enviadas ao dispositivo central por outros dispositivos na rede. A maior desvantagem desse sistema é que, se o dispositivo central for hackeado ou não estiver funcionando, toda a rede estará suscetível a mais ataques. Além disso, com o aumento dos logs de rede, o dispositivo central pode ficar sobrecarregado devido à sobrecarga excessiva. Esse IDS centralizado toma decisões independentes sobre intrusões na rede, portanto, também pode ser conhecido como um IDS independente.

No caso de um IDS distribuído, cada dispositivo na rede pode detectar e responder a intrusões. Tal IDS segue uma arquitetura hierárquica semelhante a uma árvore, onde cada nó se comunica com outros nós em uma abordagem de baixo para cima. O IDS distribuído toma decisões colaborativas em relação a um ataque detectado na rede, portanto também

é conhecido como IDS colaborativo. Em seu trabalho de pesquisa sobre IDS, Thakkar e Lohiya 2020 argumenta que alguns desafios enfrentados por um IDS distribuído são balanceamento de carga, tolerância a falhas e detecção de ameaças internas.

A próximas seções discutem os conceitos pertinentes ao desenvolvimento da presente pesquisa. Serão explorados os conceitos de ataques adversariais, treinamento adversarial e detecção OOD (com ou sem associação a amostras adversariais).

2.4 ATAQUES ADVERSARIAIS

As amostras,exemplos ou ataques adversariais são definidas com uma alteração mínima capaz de perturbar o modelo alvo com o objetivo de modificar completamente a saída do classificador Szegedy et al. 2014. Na prática, a ideia é expor uma nova superfície de ataque para aplicações baseadas em aprendizagem de máquina. O surgimento de ataques adversários motivou a pesquisa e o desenvolvimento em contramedidas a ataques adversariais, especialmente em domínios críticos de segurança He, Kim e Asghar 2023. Deve ser dito que dentre os tipos de ataques adversariais, o presente trabalho é focado nos ataques evasivos, que podem alterar o resultado do NIDS baseado em DL durante a inferência.

O estudo de Aprendizagem de Máquina Adversarial (Adversarial Machine Learning (AdvML)) é responsável por analisar as fragilidades dos sistemas baseados em modelos de inteligência artificial. Técnicas relevantes têm sido extensivamente empregadas nos últimos anos, especialmente na área de visão computacional Xu et al. 2019. Contudo, na segurança cibernética, a AdvML ainda necessita de novas contribuições, dada a presença de agentes maliciosos e a alta relevância de manter a privacidade e consistência, bem como a disponibilidade de informações. Portanto desenvolver técnicas de AdvML para defender sistemas empregados em cibersegurança são imprescindíveis, visto que as ameaças surgem e se renovam periodicamente.

A criação de amostras adversariais pode ser classificada de duas maneiras: a partir do conhecimento do atacante sobre o modelo ou com base no objetivo do atacante Jmila e Khedher 2022. Amostras baseadas no conhecimento do atacante descrevem a extensão do conhecimento do adversário sobre o sistema NIDS. Neste cenário, podemos caracterizar três níveis de riscos de ataque Han et al. 2021:

- Ataques *White-Box*: O atacante tem acesso a todas as informações sobre o modelo. Isto inclui dados de treinamento, detalhes da arquitetura do modelo, decisão e parâmetros como gradientes e função de custo.

- Ataques *Black-Box*: Este é o caso oposto, onde o atacante não tem acesso ao modelo. Portanto, ele precisa fazer várias consultas ao modelo alvo de forma que consiga inferir alguma característica que o torne apto a construir uma amostra evasiva. Esses ataques podem ser baseados em *score*, nos quais a saída do modelo alvo, probabilidades ou logits, será usada para realizar o ataque. Além disso, eles podem ser baseados em decisão, no qual as consultas têm como objetivo saber qual o rótulo de classificação dado como saída pelo modelo Wang et al. 2022. Uma outra forma de ataque considerada black-box é o ataque de transferência Demontis et al. 2018. Neste tipo de ataque, o agente malicioso treina um modelo, cria ataques adversariais white-box contra esse modelo e a partir dos ataques que conseguem evadi-lo,direciona essas amostras adversariais contra o modelo alvo. Essa amostras dependem de que o modelo do atacante tenha alguma semelhança com o modelo alvo, ou seja DL vs DL ou Machine Learning (ML) vs ML, e não há necessidade que os dados utilizados pelo atacante sejam os mesmos utilizados no treino do modelo alvo Grini et al. 2025.

- Ataques *Grey-Box*: Este cenário assume uma abordagem mais realista Jmila e Khedher 2022, onde o atacante tem um conhecimento parcial do modelo alvo e pode ter acesso limitado aos dados de treinamento. Embora não tenha as informações exatas, ele possui informações suficientes para poder atacar o sistema de DL e induzir uma falha.

Quanto aos tipos de ataques que estão condicionados ao objetivo do invasor, seja o de confundir completamente o sistema ou induzir uma previsão específica para determinadas entradas, podemos listar a seguinte categorização:

- Ataques direcionados, ou *targeted*: direcionam o algoritmo de ML para uma classe específica, ou seja, o adversário engana o classificador para prever todos os exemplos adversários como uma classe alvo específica.

- Ataque não direcionado, *untargeted*: visa classificar incorretamente a amostra de entrada para longe de sua classe original, independentemente da nova classe de

saída. Eles são mais fáceis de implementar porque mais alternativas estão disponíveis para reorientar a saída. Observe que em problemas de classificação binária, ataques direcionados e não direcionados são equivalentes.

Os ataques também podem ser compreendidos em termos do espaço das amostras Dyrmishi et al. 2022, sendo divididos em:

- Espaço do problema: no caso do NIDS, corresponde à dimensão do fluxo de pacotes e demais objetos de rede;
- Espaço de atributos: é equivalente ao vetor de atributos comumente usado para treinar/avaliar os modelos ML e DL .

2.4.1 Trabalhos anteriores em robustez adversarial via treinamento adversarial

Apesar de aumentar o custo computacional do treino e reduzir as métricas quando comparadas a um treinamento normal Zhang et al. 2019, o treinamento adversarial, Fig 4 é uma técnica considerada eficiente para robustez dos modelos a amostras evasivas , pois ao perturbar as amostras de entrada durante o treino, age como uma regularização, encoraja o modelo a aprender uma superfície de decisão mais suave, reduzindo a sensibilidade a perturbações Tsipras et al. 2019, Sinha, Namkoong e Duchi 2018, Bajaj e Vishwakarma 2023, como também induz ao aprendizado de feaures invariantes e discriminativas, o que pode contribui para uma melhor separação intra-classe Qian et al. 2022, Costa et al. 2023.

Esse tipo de treinamento consiste basicamente na geração de amostras adversárias a partir das amostras não perturbadas usadas como entrada no modelo Muhammad e Bae 2022 Fig 5.

Em Madry et al. 2017 é formulado como uma otimização min-max (Equação 2.1) para encontrar os piores exemplos possíveis (dentro de limites) forçando o modelo a aprender parâmetros que o tornem mais robusto.

$$\min_{\theta} \ \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta\in\mathcal{S}} \ \mathcal{L}(f_\theta(x+\delta), y) \right] \tag{2.1}$$

Onde cada termo da equação 2.1 representa:

θ**:** Parâmetros do modelo.

$\mathcal{D}$**:** Distribuição dos dados de treinamento.

(x, y)**:** Par de entrada e rótulo extraído da distribuição $\mathcal{D}$.

$\delta \in \mathcal{S}$**:** Perturbação adversarial dentro de um conjunto de restrições $\mathcal{S}$ (como uma bola ℓ_p).

$\mathcal{L}$**:** Função de perda (ex.: entropia cruzada).

$f_\theta(x + \delta)$**:** Saída do modelo, com parâmetros θ, após aplicação da perturbação δ à entrada x.

Algorithm 1 Treinamento Convencional

Inicialize pesos θ
for época = 1 até N **do**
  for minibatch (x, y) **do**
    $y_{\text{pred}} \leftarrow f(x; \theta)$
    $\mathcal{L} \leftarrow \text{loss}(y_{\text{pred}}, y)$
    Atualize θ com $\nabla_\theta \mathcal{L}$
  end for
end for

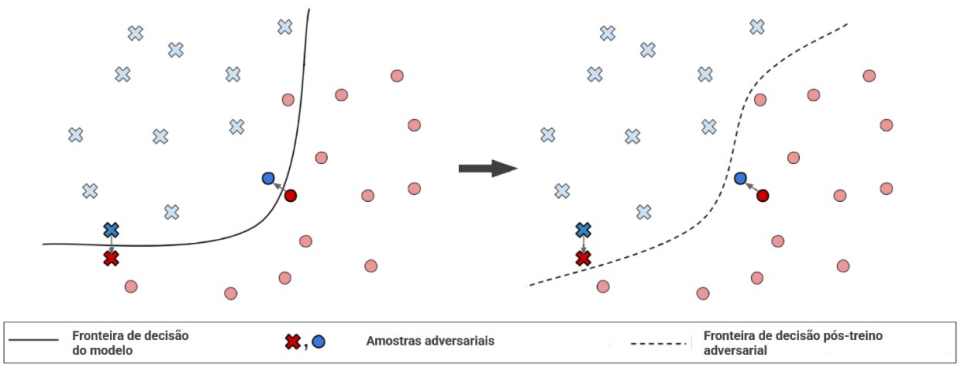
Algorithm 2 Treinamento Adversarial

Inicialize pesos θ
for época = 1 até N **do**
  for minibatch (x, y) **do**
    $g \leftarrow \nabla_x \text{loss}(f(x; \theta), y)$
    $x_{\text{adv}} \leftarrow x + \epsilon \cdot \text{sign}(g)$
    $y_{\text{pred}} \leftarrow f(x_{\text{adv}}; \theta)$
    $\mathcal{L} \leftarrow \text{loss}(y_{\text{pred}}, y)$
    Atualize θ com $\nabla_\theta \mathcal{L}$
  end for
end for

Figura 4 – Comparação entre o treinamento convencional e treino adversarial

Um dos métodos usados para treinamento adversarial é o Fast Gradient Signed Method (FGSM), proposto por Goodfellow, Shlens e Szegedy 2015. Essencialmente, os autores do

Figura 5 – Efeito do treino adversarial na fronteira de decisão



Adaptado de Taheri et al. 2020

método estudaram os efeitos da geração de amostras adversárias em modelos lineares e não lineares. Embora tenham conseguido aumentar a robustez dos modelos, testaram contra um ataque também baseado no FGSM. Madry et al. 2017 argumentam que o uso de ataques de passo único, tal como o FGSM, pode resultar em overfitting e propuseram um ataque multi-passos baseado numa otimização min-max. Seu método tem como vantagem gerar um modelo que é mais robusto do que um treinado em FGSM, embora aumente consideravelmente o tempo de treinamento, problema este que é comum a outros treinamentos adversariais.

Devido à constatação de que o erro associado à robustez pode ser causado tanto por erro do treinamento associado às amostras usadas, quanto pelas perturbações adicionadas que modificam as amostras, Zhang et al. 2019 desenvolveram um método que possibilita um melhor controle do trade-off acurácia-robustez baseado no valor de um hiperparâmetro na função de custo por eles proposta. Os autores combinaram entropia cruzada e divergência Kullback-Leibler e essa abordagem na função de custo se mostrou promissora e ainda é utilizada em treinamentos adversariais Wang et al. 2019,Wang et al. 2020 e Nandi et al. 2023.

Indo numa direção diferente, Wang et al. 2019 perceberam diferenças na robustez ao considerar amostras não atacadas que foram classificadas erroneamente pelo modelo. Em particular, os autores perceberam que a minimização feita no treinamento adversarial min-max é sensível a estes erros. O tipo de treinamento que eles apresentam utiliza uma função de custo com uma regularização que permite diferenciar amostras corretamente classificadas das classificadas erroneamente.

De acordo com Nandi et al. 2023, os treinamentos adversários disponíveis são restritos a perturbações muito pequenas, o que poderia gerar vulnerabilidade diante de perturbações um pouco mais significativas. Para resolver o problema, propuseram um treinamento que é robusto a um intervalo de perturbações baseado em uma combinação de ruído gaussiano e ruído uniforme, o qual gera uma faixa que resulta na robustez do modelo.

2.5 DETECÇÃO DE AMOSTRAS OOD

A razão fundamental de usar métodos para detecção de OOD é diferenciar entre dados ID, ou seja, dados da distribuição de treinamento, e os dados fora de distribuição, que podem se originar de uma fonte ou contexto diferente do encontrado durante o treinamento.

Um NIDS confiável deve produzir previsões acuradas e, ao detectar exemplos desconhecidos, rejeitá-los. No entanto, a maioria dos modelos DL são treinados com premissas "mundo fechado", ou seja, a distribuição dos dados na inferência é a mesma dos dados de treinamento. Porém, tal fenômeno não costuma ser observado nos modelos em produção Yang et al. 2024. Na verdade, após a implantação do modelo, ocorrem as amostras consideradas OOD, as quais degradam a performance do modelo, diminuindo assim sua confiabilidade Zhou et al. 2021.

Conforme discutido por Yang et al. 2024, as modificações na distribuição dos dados podem ser causadas por:

- **Mudanças semânticas**: onde há classes diferentes das vistas no treinamento;
- **Mudanças de covariável, ou variável preditora**: as quais são resultantes de diferenças entre domínios, refletindo mudanças nas propriedades estatísticas dos dados na inferência. Essas mudanças podem incluir mudanças na escala, estilo ou padrões dos dados. Karunanayake et al. 2025 ainda cria uma subdivisão em três categorias:

i) detecção de anomalia sensorial/outlier;

ii) robustez adversarial;

iii) generalização de domínio.

Embora as técnicas ii e iii relacionadas a mudanças de covariável compartilhem o objetivo de melhorar a generalização do modelo, ao aplicar a NIDS elas variam em sua resposta à intenção maliciosa causada por mudanças na distribuição.

2.5.1 Robustez adversarial em OOD

Sehwag et al. 2019 testaram a robustez de um modelo OOD a ataques adversários evasivos. Os autores concluiram que o método em questão não detectava as amostras adversárias. No entanto, apenas recentemente foram propostos modelos focados em OOD com robustez a amostras adversariais.

A partir de distribuições gaussianas condicionadas por classe, Lee et al. 2018 criaram um método para pontuações de confiança por distância Mahalanobis. Este método con-

segue detectar amostras adversárias e OOD.No presente trabalho, a detecção de OOD e amostras adversariais é feita com base neste método.

Em Malinin e Gales 2019, os autores propuseram uma função de custo para Prior Networks (PN). Em sua pesquisa perceberam que a função divergência Kullback-Leibler reversa é mais apropriada para lidar com as estimativas de incerteza feitas pelas PN. Entretanto, para além da quantificação de incerteza nesse tipo de rede, a abordagem dos autores permite OOD e detecção de amostras adversárias com norma ℓ_2 e ℓ_∞.

Em Arjovsky et al. 2019 é desenvolvido o método Invariant Risk Minimization (IRM) para estimar correlações invariantes entre múltiplas distribuições durante o treino. Porém Xin et al. 2023 afirmam que esse método não é suficiente para detecção de OOD. Uma vez que IRM consegue detectar mudanças na distribuição, mas é inapropriado para mudanças de diversidade na amostra, os autores [Xin et al. 2023] então propõem uma combinação de treino adversário e IRM para detecção de ambos os tipos de mudança.

Wang et al. 2022 usaram o universal attack Moosavi-Dezfooli et al. 2016 para melhorar detecção de amostras adversárias. Especificamente, seu método gera amostras adversárias de baixo posto matricial numa imagem, o que resultou em detecção de OOD e robustez exclusivamente a ataques de norma ℓ_2. Particularmente, a abordagem dos autores resultou numa melhor generalização para grandes perturbações.

Nas seções seguintes serão apresentados resumidamente os trabalhos publicados que embasam esta dissertação. Sendo assim são abordados aqui: detecção de intrusão em redes com e sem inclusão de robustez adversarial e intrusão em redes com uso de OOD.

2.6 DETECÇÃO DE INTRUSÃO EM REDES

Sistemas IDS representam uma camada defensiva adicional contra acessos não autorizados em redes e computadores. Eles complementam outras medidas de segurança, como controle de acesso e procedimentos de autenticação, formando um sistema de proteção integrado. No contexto de IDS com arquitetura baseada em DL, distinguem-se três abordagens principais: aprendizado único, aprendizagem em comitê e aprendizado híbrido. Geralmente, essas metodologias são aplicadas em classificadores que têm a função de discernir entre um fluxo de dados normal e um ataque.

- **Aprendizagem única**: Um único algoritmo de aprendizagem de máquina é usado

como NIDS ou HIDS.

- **Aprendizagem em comitê**: A combinação de vários modelos é chamada de comitê. Classificadores combinados geram melhores resultados, comparados à metodologia de aprendizagem única. Pode ser usado o método "voto majoritário"para obter melhor performance na classificação, como também podem ser utilizadas outras técnicas como *bagging* e *boosting*, onde são feitas reamostragens no treinamento.

- **Classificadores híbridos**: Combina métodos baseados em assinaturas e ML para melhorar o desempenho classificatório. Há dois componentes funcionais, onde o primeiro usa os dados de entrada e produz um resultado intermediário que será usado pelo segundo componente, o qual será responsável pela classificação.

Tang et al. 2016 propuseram o uso de uma Multi-Layer Perceptron (MLP) com 5 camadas para NIDS em Sofware-Defined Networking (SDN). Utilizaram o daset NSL-KDD [Tavallaee et al. 2009] e pré-processamento manual para redução de features. Com a redução na quantidade de features, onde apenas 6 das 41 features originais foram utilizadas. Segundo os autores o modelo resultante é leve e apropriado para redes tipo SDN, em razão da quantidade de features. Eles concluem que seus resultados mostram a viabilidade de usos do DL para detecção de anomalias.

Potluri e Diedrich 2016 usaram um modelo DL para NIDS. Usando o dataset NSL-KDD como *benchmark*, os autores agruparam as classes para um modelo binário. Em particular, obtiveram métricas como Precisão e ROC-AUC em torno de 99%. Ao tentar abordar o problema via classificação multiclasse, tiveram dificuldade com as classes minoritárias. Deve ser dito que o problema do desbalanceamento em datasets ainda é um desafio para muitos modelos de ML e DL Chen et al. 2024.

Kang e Kang 2016 propuseram melhorar a segurança de redes veiculares através de um IDS baseado em DL, com um foco em Controller Area Network (CAN). Os autores treinaram o modelo com dados a partir dos vetores de características do CAN. Para melhorar as métricas classificatórias Acurácia, Precisão e Recall, eles utilizaram para pré-processamento uma rede tipo Deep Belief. Usando uma MLP como algoritmo de classificação, os autores simularam dados CAN com 200 000 pacotes. A acurácia do modelo ficou em 97.8% e falso positivos em 1.6%. Segundo os autores, o tempo de extração

ficou em média 8 microssegundos. Segundo os autores esta baixa latência torna o modelo apropriado para aplicações de tempo real.

O trabalho de [Zhou et al. 2018] foca em detectar ataques em *smart grids*. Os autores usaram um *Stacked Denoising Autoencoder* como classificador DL e compararam seu desempenho com os métodos: Random Forest, K-Nearest Neighbors (KNN) e Regressão Linear. O modelo proposto obteve as melhores métricas – Acurácia e F1-Score – segundo os autores, pode ser aplicado a situações onde se exige classificação em tempo real.

Os autores em [Darem et al. 2021] usam um DL com treinamento semi-supervisionado para detecção de malwares que se ofuscam. Levaram em consideração features de rede e host, transformando os dados utilizados em imagens de 8 bits e usaram algumas técnicas como salientar segmentos e intensidade de pixels, combinando com contagem de caracteres dos arquivos ASM dos malwares. Em seguida, fizeram uma seleção de features usando Random Forest. O modelo utilizado é uma combinação entre uma rede neural convolucional e um comitê *xgboost*. Os autores reportaram resultados em torno de 99% para métricas como acurácia a ROC-AUC.

[Maseer et al. 2021] realizaram um benchmark de modelos MLP, rede convolucional e mapa auto-organizável. Entre os modelos de ML foram incluídos na comparação: árvore de decisão, KNN, k-means e naive bayes. Os modelos foram treinados com o conjunto de dados CIC IDS2017, utilizando 38 das 80 features presentes no dataset. Os resultados mostram que os modelos com melhores desempenho foram árvores de decisão e KNN, em particular, obtiveram 99% na acurácia. Já os modelos DL obtiveram performance comparável aos modelos ML com 99% na acurácia, excetuando o mapa auto organizável, o qual alcançou 59% de acurácia. Os autores concluem que, embora os algoritmos de ML sejam mais rápidos no treinamento, eles pretendem pesquisar melhorias para modelos DL em termos acurácia de detecção.

2.7 ROBUSTEZ ADVERSARIAL EM NIDS

Modelos de DL têm sido amplamente aplicados à cibersegurança devido às suas robustas capacidades de generalização. No entanto, a vulnerabilidade desses modelos a ataques adversariais representa um ponto fraco em sistemas de defesa que dependem desses algoritmos. Consequentemente, há um interesse crescente em desenvolver métodos para mitigar ataques a modelos de DL em IDS.

Usando dados do CIC-IDS 2017 para *benchmark*, Pawlicki, Choraś e Kozik 2020 propuseram um sistema com dois classificadores, sendo o primeiro voltado para o fluxo normal ou ataques e o segundo para discriminar fluxo benigno ou amostras adversariais. A partir da extração da ativação das camadas do classificador IDS, os autores rotularam um subconjunto de dados, contendo a informação das ativações do primeiro classificador, e treinaram o segundo classificador. O método obteve resultados acima de 95% para as métricas Acurária e F1-Score com as amostras adversariais.

Um modelo de reconstrução por observação parcial foi proposto por Hashemi e Keller 2020. Especificamente, a abordagem foca em ataques adversariais mais realistas em termos de manipulação das features. Utilizando um *denoising autoencoder* para detecção de anomalias, os autores fizeram uma replicação de amostras, por 100 vezes, com mascaramento de até 75%. Em seguida, foi efetuada uma seleção das amostras com menor erro de reconstrução obtido pelo autoencoder. Por fim, os autores estimam o limiar no qual a amostra é adversarial ou não a partir do score de reconstrução do agrupamento dos dados.

Em [Peng et al. 2020], os autores usam uma Generative Adversarial Netork (GAN) [Goodfellow et al. 2021] para detecção pré-IDS a partir do erro de reconstrução. Baseado numa arquitetura GAN bidirecional e estimativas de erros por *feature matching*. A proposta usa as saídas em valores absolutos de uma função que estima o custo do *encoder*-gerador e *encoder*-discriminador juntamente com uma função de score criada pelos autores. A depender do score, o modelo seleciona quais amostras irão para o IDS ou quais seriam descartadas. Deixam para um trabalho futuro, um método adaptativo para estabelecer limiar.

Em Zhang, Costa-Pérez e Patras 2020 é utilizado um comitê de três classificadores com treinamento adversarial Projected Gradient Descent (PGD), combinado com um encoder contrastivo [Chen, Carlini e Wagner 2019] para detectar consultas associadas a ataques black-box. Testaram com os dados CIC IDS2018 e os ataques adversariais: FGSM,PGD, opt attack Liu, Sun e Li 2020, hopskipjump Chen e Jordan 2019 e boundary Brendel, Rauber e Bethge 2017. Uma das vantagens desse método é que ao detectar as consultas dos ataques black-box, isso diminuiu a capacidade de perturbação feita pelos métodos adversariais.

Uma abordagem baseada em detecção foi proposta por Wang et al. 2022, utilizando um sistema de decisão baseado na variedade topológica dos dados na inferência. Os autores

notaram que amostras adversárias ocorrem próximo a variedade que as originaram, portanto próximos à fronteira de decisão do modelo, assim os ataques geram as perturbações imperceptíveis. Com o CIC-IDS 2017 e NSL–KDD para seus experimentos, utilizaram os ataques FGSM, Basic Iterative Method (BIM) e Carlini-Wagner (CW). Uma das vantagens dessa abordagem é amenizar o trade-off entre acurácia-robustez mencionado em Zhang et al. 2019 uma vez que não usa treino adversarial, contudo o trabalho não mostra se o método detecta ataques black-box.

Chauhan e Shah-Heydari 2020 propuseram um método baseado em GAN para criar versões adversariais do ataque DDoS a partir do CIC IDS2017. Seu método tem a vantagem de criar ataques realistas no espaço de features. Para tornar o modelo IDS baseado em DL mais robusto a este ataque, fizeram o treinamento a partir do modelo generativo atacante, o que aumentou em até 79% a capacidade de detecção dos ataques adversariais criados.

2.8 OOD EM NIDS

Corsini e Yang 2023 realizaram um estudo comparativo entre métodos OOD em modelos DL para NIDS. Seu trabalho avaliou cinco métodos de OOD num classificador MLP primeiro com e depois sem função de custo contrastiva. Para validar a eficácia do método, utilizaram dados CIC IDS2017. Já para simular dados OOD, utilizaram o CIC IDS2018 [1]. Segundo os autores, houve melhora na detecção para os modelos MLP testados independentemente da função de custo. Além disso, o OOD possibilitou detectar ataques que não foram percebidos pelo NIDS em razão da mudança de distribuição.

Os autores em [Ceccarelli e Zoppi 2023] desenvolveram um NIDS com ausência de conhecimento prévio de ataques. Os ataques foram criados a partir das imperfeições dos dados normais com técnicas de aumento de dados usando GANs para dados tabulares. Os autores usaram o CIC IDS2018 e o ADFA para avaliação. Os resultados iniciais do seu NIDS baseado em DL, que detecta anomalias, superaram os do outro modelo usado como baseline que utiliza apenas detecção de baseada em classe única. Assim, o modelo final foi um detector binário seguido de outro detector para OOD. De forma resumida, a classificação binária no geral ficou entre 98% e 99%. Já para o modelo baseado em classe única, ficou em 95%.

[1] https://www.unb.ca/cic/datasets/ids-2018.html.

Zhao et al. 2022 propuseram um NIDS voltado para uma CAN em veículos. Seu método consegue identificar ataques conhecidos e desconhecidos. A abordagem por eles desenvolvida tem como base uma GAN adicionando um classificador auxiliar e detecção de OOD. Os autores testaram 4 arquiteturas baseadas em redes convolucionais, todas elas juntamente com o detector OOD, porém três delas com uso da GAN no treino. Um dos modelos obteve bons resultados nas métricas de avaliação (Precisão, Recall e F1-Score) e segundo os autores foi eficiente computacionalmente com tempo de inferência da ordem de 0.2 milissegundos. O estudo conclui que a combinação de treino com GAN, classificador binário e OOD é eficiente na defesa para redes veiculares, podendo inclusive ser implementado em sistemas embarcados.

Wong et al. 2023 comparam 4 modelos DL MLP com outros 4 modelos DL bayesianos, e quantificam incerteza com o método Monte Carlo Hamiltoniano. Segundo os autores, mesmo os modelos MLP não são confiáveis quando se trata de ataques *zero-day*. Ao usar quantificação de incerteza para discriminar ID e OOD, eles conseguiram uma melhora de aproximadamente 87% na detecção de ataques não vistos no treinamento. O trabalho conclui que apesar do custo computacional dos modelos bayesianos, a melhora na detecção foi considerada satisfatória para ataques *zero-day* em versões modificadas dos datasets CIC IDS2017 e UNSW-NB15.

2.9 ATAQUES ADVERSARIAIS EM OOD

Liang et al. 2022 aplicaram *few-shot learning* para NIDS Internet of Things (IoT) em ambientes industriais. Especificamente, os autores utilizaram os datasets NSL-KDD e CIC IDS2017, e o modelo proposto sendo capaz de lidar com o desbalanceamento de ambos. O método densenvolvido mostrou ser capaz de detectar comportamento malicioso com poucos dados disponíveis. A partir de modificações feitas nos datasets, os autores ainda simularam OOD e sua abordagem obteve métrica F1 score para ID entre 95% e 98%, além de um ROC-AUC em 98% para OOD. Testaram, inclusive, a resiliência do modelo a ataques adversariais baseados em transferência e concluíram que o modelo também foi robusto a esse tipo de ataque, pois as perturbações causaddas no modelo proposto foram inferiores a 1% (ROC-AUC) para as diferentes intensidades do ataque.

3 METODOLOGIA

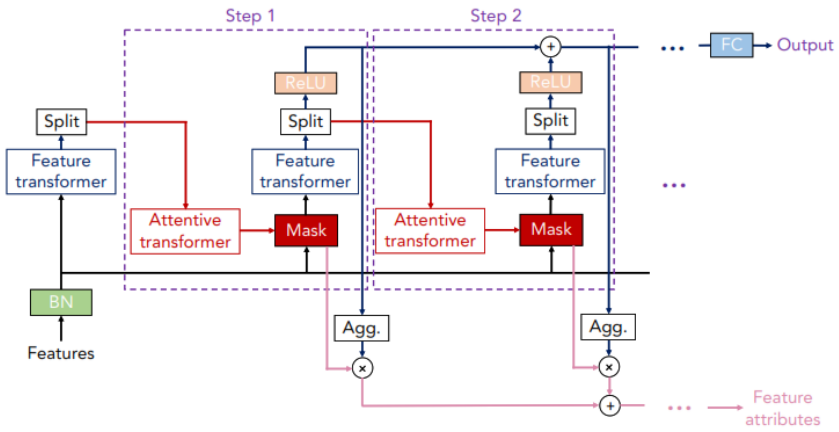
Este capítulo apresenta a metodologia utilizada no desenvolvimento, treinamento e avaliação da solução de detecção de intrusão proposta. O desenvolvimento foca em treinar um modelo deep learning de forma a torná-lo robusto tanto a ataques adversariais quanto à presença de dados fora da distribuição original.

3.1 TABNET

Embora redes neurais profundas se destaquem em dados de imagem, texto e áudio, elas não representam necessariamente o estado da arte quando se trata de dados tabulares. Geralmente, comitês como CatBoost Dorogush, Ershov e Gulin 2018 e XGBoost Chen e Guestrin 2016 superam as redes neurais no tratamento de dados tabulares. Por essa razão, têm sido propostas arquiteturas de Deep Learning (DL) para preencher essa lacuna. Uma dessas arquiteturas é a TabNet Arik e Pfister 2019, que emprega um mecanismo de atenção para selecionar atributos relevantes (Fig 6). Tem arquitetura flexível para aprendizado supervisionado e auto-supervisionado e possui a vantagem de ser interpretável das formas abaixo:

- **Localmente** Al e Sağıroğlu 2025: Quando é possível explicar ou entender as razões por trás de uma predição individual (ou um pequeno conjunto de predições). Isso significa analisar como os inputs específicos (por exemplo, features de uma amostra) influenciaram a saída do modelo para aquela instância;
- **Globalmente** Arreche et al. 2024: Quando conseguimos descrever o comportamento geral do modelo em todo o espaço de inputs. Isso envolve entender padrões, regras ou tendências aprendidas pelo modelo. No caso da TabNet essa forma de interpretação é via importância de features no conjundo de dados;

Figura 6 – Camadas da TabNet



Camadas: **Feature Transformer**: Bloco com camadas totalmente conectadas e de normalização em lote; **Attentive Transformer**: Gera uma máscara de atenção que pondera os atributos a serem usados na próxima etapa; **Step** n: Etapa que foca em diferentes subconjuntos de atributos, promovendo diversidade de representação e esparsidade. Fonte: Arik e Pfister 2019

3.2 CONSIDERAÇÕES SOBRE A TABNET E INTERPRETABILIDADE

Embora a TabNet seja uma arquitetura baseada em atenção e pode gerar interpretabilidade de resultados em forma de importância de features, isso não é necessariamente aplicável em caso de treinamento adversarial Noack et al. 2020, portanto após o treino com amostras adversariais tem-se um modelo robusto a alguns ataques ou até mesmo a um tipo de norma ℓ_p Madry et al. 2017,Nandi et al. 2023 sacrificando a confiança na interpretabilidade Noack et al. 2020.

O trabalho de Si et al. 2023 destaca limitações estruturais e funcionais do uso de atenção em cenários tabulares. Segundo eles os pesos de atenção da TabNet frequentemente resultam em distribuições densas, sensíveis a pequenas perturbações, e que não necessariamente refletem a real importância causal das variáveis de entrada. Deve ser dito que já vem sendo propostos modelos de treinamento adversarial que tornam a interpretabilidade baseada em atenção robusta Kitada e Iyatomi 2021, porém este método conseguiu em geral melhoras inferiores a 4% para as métricas utilizadas nos experimentos, o que não é apropriado para ataques fortes, ou seja, ataques adversariais os quais se mantém imperceptíveis e enganam o modelo com alta eficácia, tais como os que foram testados na presente dissertação.

3.3 IMPERCEPTIBILIDADE DE ATAQUES ADVERSARIAIS EM DADOS TABULARES

A maioria dos métodos para ataque adversarial foi desenvolvida para dados de imagem, sua aplicação a dados tabulares requer consideração das características únicas destes últimos. Além dos critérios baseados na distância, é necessário levar em conta a natureza específica dos dados tabulares ao abordar a imperceptibilidade de ataques adversários.

No trabalho de He et al. 2024, onde foram estudados ataques adversariais em dados tabulares, são enumerados os seguintes critérios para imperceptibilidade:

- Minimização da perturbação das features: A amostra criada pelo ataque deve ser o mais próxima possível dos dados de entrada, e também nem todas as features devem ser modificadas.

- Preservação da distribuição de dados estatísticos: Espera-se que os ataques estejam alinhados com a distribuição dos dados de entrada. Exemplos adversários que se desviam significativamente das propriedades estatísticas originais têm maior chance de serem detectados pelo modelo.

- Perturbação de características numa faixa estreita: Em dados tabulares, cada feature tipicamente exibe uma distribuição única. Quando perturbações são aplicadas entre features, features com distribuções mais estreitas sofrem mais impacto que as features com distribução ampla. Portanto, para o ataque ser imperceptível, as perturbações devem evitar alterar features com distribução estreita.

- Preservação da semântica das características: Em dados tabulares, cada feature geralmente tem uma semântica definida e valores válidos. Porém, as perturbações introduzidas pelos ataques adversariais podem alterar a semântica das features ou modificar os valores para além da faixa de valores válidos (por exemplo, um campo de idade que ao invés de 25 esteja 150). Logo, para garantir a imperceptibilidade dos ataques em dados tabulares, a semântica precisa ser preservada.

- Preservação das interdependências de características: Dados tabulares podem conter features interdependentes e para o ataque ser imperceptível features interdependentes devem ser alteradas levando em consideração suas relações.

Ainda em He et al. 2024 também são usadas as seguintes métricas de maneira a observar as características que indicam imperceptibilidade.

3.3.1 Abordagem baseada em Desvio

Os autores em Lee et al. 2018 sugerem que ataques adversariais são um exemplo especial de OOD, porém com o propósito de enganar um modelo. Apesar de exemplos adversariais não serem considerados representativos da distribuição real em modelos preditivos, uma entrada perturbada deve ser o mais semelhante possível à maioria das entradas originais, para preservar a distribuição estatística dos dados.

Na presente dissertação, a métrica utilizada para medir o Desvio será a *distância Wasserstein*. De acordo com Wu, Wang e Yu 2020 tem como vantagem conseguir capturar a informação geométrica no espaço dos dados , ou seja, da importância em como a "massa" distribucional das features se modifica. Portanto é uma métrica usada aqui para quantificar o quanto a distribuição das features atacadas foram modificadas.

3.3.2 Abordagem baseada em Proximidade

A partir do critério de minimização da perturbação de features, um bom exemplo adversarial introduzirá mudanças mínimas, que podem ser quantificadas mantendo a menor distância possível do vetor de features original. Empregamos a norma ℓ_p para medir a distância de perturbação. Para medir a proximidade, serão usadas métricas de magnitude de um vetor em espaços n-dimesionais. No presente caso, são elas: distância em linha reta (distância ℓ_2) e diferença máxima de características (distância ℓ_∞).

3.3.3 Avaliação de imperceptibilidade adversarial em dados tabulares

Neste trabalho o desempenho dos classificadores de fluxo de rede serão avaliados com as métricas: Precisão, Recall e ROC-AUC. Para o classificador/detector de out-of-distribution será ROC-AUC. Desse modo pode-se ter a noção da dos Falsos positivos/negativos dos modelos.

A avaliação de imperceptibilidade será feita com as métricas de distância: L_2, L_∞ e Wasserstein. Sendo porém a métrica de distância L_2 ou L_∞ de acordo com a ℓ_p do ata-

que, conforme tabela 1, desse modo é avaliado a distorção causada pela norma que o ataque usa para criar as amostras adversariais. Os resultados estarão presentes no Anexo A. As métricas de distância aqui usadas refletem a preservação da distribuição. Sobre a preservação de semântica das características, minimização da perturbação e preservação de interdependência serão realizadas diretamente nos ataques com base na máscara binária usada no método adversarial. Deve ser dito que os trabalhos com dados tabulares que abordam métricas de imperceptibilidade He et al. 2024 e Mathov et al. 2022, não estabelecem um método para estimar limiar, desse modo foi usada inspeção manual com a biblioteca Pandas (Pandas development team 2020) a partir da diferença absoluta entre amostas clean e amostras perturbadas. Portanto foi considerado imperceptível o ataque que a diferença absoluta seja menor que 0.5 para distância L_2 e L_∞, tal como ocorre nos trabalhos citados. Pela mesma razão,o limiar usado para distância Wasserstein foi para valores $\leq 5 \times 10^{-2}$.

Tabela 1 – Uso de métrica de distância com base na norma do ataque

Norma	ℓ_2	ℓ_∞
Métrica	L_2 e Wasserstein	L_∞ e Wasserstein

3.4 MÁSCARA BINÁRIA

De forma a gerar amostras com as restrições características de dados tabulares para segurança de redes, estre trabalho adapta os ataques utilizados e o treino adversarial a partir de uma máscara binária. Ou seja, o treinamento adversarial utilizado e o código dos ataques CW e SignOPT (Algoritmo 3)) foram modificados de maneira a gerar amostras levando em consideração as features que podem ser modificadas e as que não devem ser modificadas, desse modo podem ser geradas amostras semanticamente válidas Kuppa et al. 2019 e Zhang, Costa-Pérez e Patras 2020 . Os ataques PGD e HopSkipJump não foram alterados pois a biblioteca Adversarial Robustness Toolbox Nicolae et al. 2018, utilizada para gerar as amostras adversariais, já os disponibiliza com suporte a máscaras binárias. Nas tabelas 2 e 3 são mostradas as features dos datasets, em negrito as features que os ataques podem alterar.

O uso da máscara binária ajudará em:

- manter a semântica das características: Ao modificar apenas features que não inter-

ferem na validação de tráfego de rede, desse modo a semântica "benigno"ou "malicioso"se mantém.

- preservação de interdependência: Ao criar a máscara, torna-se opcional atacar features interdependentes. No presente trabalho, a opção foi mantê-las sem perturbação, pois facilita manter a semântica e remove a necessidade de filtrar amostras inválidas.

Deve ser lembrado que outra característica importante é a minimização da perturbação. Porém a necessidade de perturbação mínima para imperceptibilidade será a partir das características dos ataques adversariais, que de acordo com seus autores são todos otimizados para perturbação mínima de features.

Tabela 2 – Features do UNSW-NB15

Tipo de feature	**Features**
Conexões de rede	state, dur, sbytes, dbytes, sttl, dttl, sloss, Dloss, service,sload, dload, spkts, dpkts
Pacotes	swin, dwin, stcpb, dtcpb, smeansz, dmeansz, Trans_depth, res_bdy_len
Fluxo	srcip, sport, dstip, dsport, proto
Temporais	sjit, djit, stime, ltime, sintpkt, dintpkt, tcprtt, Synack,ackdat
Adicionadas	**is_sm_ips_ports**, **ct_state_ttl**, **ct_flw_http_mthd**, **is_ftp_login**, **ct_ftp_cmd**, **ct_srv_src**, **ct_srv_dst**, **ct_dst_ltm**, **ct_src_ltm**, **ct_src_dport_ltm**, **ct_dst_sport_ltm**,**ct_dst_src_ltm**

Tabela 3 – Features do CIC IDS2017

Tipo de feature	**Features**
Fluxo	Source IP, Destination IP, Source Port, Destination Port, Protocol
Temporais	Flow Duration, Flow Bytes/s, Flow Packets/s
Estatística de pacotes	**Total Fwd/Bwd Packets**, **Fwd/Bwd Packet Length Mean/Min/Max/Std**
Tempo entre pacotes	**Fwd/Bwd IAT Mean/Min/Max/Std (Inter-Arrival Time)**
TCP Flags	FIN Flag Count, SYN Flag Count, PSH Flag Count, URG Flag Count, ACK Flag Count
Janelas TCP	Init_Win_bytes_forward, Init_Win_bytes_backward
Tempo Atividade/Ocioso	**Active Mean/Std/Max**, **Idle Mean/Std/Max**
Razões/Proporções	**Down/Up Ratio**, **Average Packet Size**, **Fwd/Bwd Packets/s**

Algorithm 3 Pseudo-código para ataque adversarial com máscara binária integrada

Require: Modelo f, exemplo original $\mathbf{x}$, rótulo y, máscara $\mathbf{m} \in \{0,1\}^d$, hiperparâmetros do ataque
Ensure: Exemplo adversarial $\mathbf{x}_{adv}$ restrito a $\mathbf{m}$

Inicialização:
$\mathbf{x}_{adv} \leftarrow \mathbf{x}$
$\delta \leftarrow \mathbf{0}$ ▷ Perturbação inicial
if ataque usa **busca binária then**
    Defina limites l e u
    while $u - l >$ tolerância **do**
        $c \leftarrow (l + u)/2$
        $\delta \leftarrow$ OtimizarPerturbacao$(f, \mathbf{x}, y, c, \mathbf{m})$
        if $\mathbf{x} + \delta$ é adversarial **then**
            $u \leftarrow c$
        else
            $l \leftarrow c$
        end if
    end while
else if ataque usa **gradiente then**
    for $t = 1$ **to** iterações **do**
        Calcule gradiente $\nabla_\delta \mathcal{L}$
        $\delta \leftarrow \delta + \alpha \cdot \text{sign}(\nabla_\delta \mathcal{L}) \odot \mathbf{m}$
        Projete δ em ϵ-bola
    end for
end if
$\mathbf{x}_{adv} \leftarrow \mathbf{x} + \delta$
return $\mathbf{x}_{adv}$

Algorithm 4 OtimizarPerturbacao

function OTIMIZARPERTURBACAO$(f, \mathbf{x}, y, c, \mathbf{m})$
    Minimize $\mathcal{L}(\delta) = \|\delta\|_2 + c \cdot$ perda_cls
    Sujeito a $\delta \odot (1 - \mathbf{m}) = \mathbf{0}$
    Use L-BFGS ou SGD
    return δ
end function

Exemplos de Aplicação dos Algoritmos 3 e 4

- **Carlini-Wagner (CW):** Modifique a otimização para que o gradiente ∇_δ seja multiplicado por $\mathbf{m}$ antes de cada atualização.
- **Sign-OPT:** Ao calcular a direção de busca s (sign do gradiente), faça $s \leftarrow s \odot \mathbf{m}$.

Observações

- A máscara $\mathbf{m}$ atua como um *gate* binário: pixels com $\mathbf{m}_i = 0$ são **inalterados**.
- A restrição é aplicada **durante** a otimização, não apenas no resultado final.

Tabela 4 – Descrição das variáveis e símbolos usados no pseudocódigo.

Símbolo	Tipo	Descrição
f	Modelo	Função que representa o modelo de classificação (ex: rede neural).
$\mathbf{x}$	Vetor	Exemplo de entrada original (ex: imagem, vetor de features).
y	Escalar	Rótulo verdadeiro associado a $\mathbf{x}$.
$\mathbf{m}$	Vetor binário	Máscara que define quais elementos de $\mathbf{x}$ podem ser perturbados ($\mathbf{m}_i = 1$) ou não ($\mathbf{m}_i = 0$).
δ	Vetor	Perturbação adversarial, restrita a $\mathbf{m} \odot \delta = \delta$.
$\mathbf{x}_{adv}$	Vetor	Exemplo adversarial gerado por $\mathbf{x} + \delta$.
α	Escalar	Taxa de aprendizado (passo da otimização).
c	Escalar	Hiperparâmetro de trade-off entre magnitude da perturbação e sucesso do ataque (usado em Carlini-Wagner).
ℓ_p	Norma	Norma utilizada para medir a magnitude da perturbação (ex: ℓ_2, ℓ_∞).
$\odot$	Operador	Produto elemento a elemento
$\nabla_\delta \mathcal{L}$	Vetor	Gradiente da função de perda em relação a δ.

NOTAS ADICIONAIS

- **Restrição da máscara**: A operação $\delta \odot (1 - \mathbf{m}) = \mathbf{0}$ garante que apenas os elementos onde $\mathbf{m}_i = 1$ sejam perturbados.

- **Hiperparâmetros**: Valores como α e c são tipicamente ajustados empiricamente.
- **Implementação**: Em frameworks como PyTorch, a máscara é aplicada via multiplicação elementar (ex: `delta *= mask`).

3.5 TREINO ADVERSARIAL

Como primeira defesa contra ataques adversariais será adaptado o treino descrito em Nandi et al. 2023 (Multiple Perturbation Bounds (MPB)). Este treino adversarial consiste em usar uma regularização de similaridade no treino através de ruído gaussiano e uniforme, seguido de uma regularização na função de custo entropia cruzada. A máscara binária também será usada neste treino de maneira que as perturbações mantenham as restrições de dados tabulares.

Regularizador de Similaridade

O objetivo do regularizador é alinhar as predições do classificador sob o ruído de treinamento (NU) com as predições sob os ruídos usados na certificação (Normal e Uniforme). Isso garante consistência durante o treinamento, mesmo quando os ruídos são diferentes.

$$\mathcal{R}_s = \mathrm{KL}(f(x + \mathbf{m} \cdot \mathrm{NU}) \,\|\, f(x + \mathbf{m} \cdot \mathcal{N})) + \mathrm{KL}(f(x + \mathbf{m} \cdot \mathrm{NU}) \,\|\, f(x + \mathbf{m} \cdot \mathcal{U})) \quad (3.1)$$

- $\mathbf{m}$: Máscara binária
- $f(\cdot)$: Classificador base que retorna um vetor de probabilidades de dimensão K
- NU: Ruído da distribuição Normal-Uniforme
- $\mathcal{N} \sim \mathcal{N}(0, \sigma^2)$: Ruído Gaussiano
- $\mathcal{U} \sim \mathcal{U}(-\sqrt{3}\sigma_u, \sqrt{3}\sigma_u)$: Ruído Uniforme
- $\mathrm{KL}(P \,\|\, Q)$: Divergência de Kullback-Leibler entre distribuições P e Q

Função de Perda Total

A função de perda total combina a entropia cruzada com o regularizador de similaridade, desse modo essa formulação permite que o modelo aprenda a ser robusto simultaneamente a perturbações ℓ_1,ℓ_2 e ℓ_∞ , algo essencial para aplicações reais onde múltiplos tipos de ataques podem ocorrer:

$$\mathcal{L} := \mathcal{L}_{\text{CE}}(f(x + \mathbf{m} \cdot \text{NU}), y) + \beta \cdot \mathcal{R}_s \tag{3.2}$$

- $\mathbf{m}$: Máscara binária
- $\mathcal{L}_{\text{CE}}$: Perda de entropia cruzada entre a predição e o rótulo verdadeiro y
- β: Hiperparâmetro que controla o peso do regularizador

3.6 DETECÇÃO DE OOD

Para a segunda etapa da defesa o detector Mahalanobis proposto por Lee et al. 2018 será modificado a partir da normalização presente em Mueller e Hein 2025, uma vez que esta alteração não altera o tempo de treino e evita que os dados de entrada durante a inferência precisem seguir uma distribuição gaussiana.

Algorithm 5 Score de confiança para OOD utilizando distância Mahalanobis

Require: Amostra de teste x, ativações da penúltima camada α_ℓ, magnitude do ruído ϵ, e parâmetros por classe $\{\mu_{b_\ell,c}, \Sigma_{b_\ell}\ \forall \ell, c\}$

Inicialização do vetor score: $M(x) = [M_\ell\ \forall \ell]$
Normalização das features da penúltima camada: $\hat{f}(x) = \dfrac{f(x)}{\|f(x)\|_2}$
for each layer $\ell \in \{1, \ldots, L\}$ **do**
    Distância Mahalanobis: $D_c = (\hat{f}_\ell(x) - \mu_{b_\ell,c})^\top \Sigma_{b_\ell}^{-1} (\hat{f}_\ell(x) - \mu_{b_\ell,c})$
    Cálculo de classe mais próxima: $\hat{c} = \arg\min_c D_c$
    Adição de perturbação: $x_b = x - \epsilon \cdot \text{sign}\,(\nabla_x D_{\hat{c}})$
    Recálculo das distâncias com entrada perturbada: $D'_c = (\hat{f}_\ell(x_b) - \mu_{b_\ell,c})^\top \Sigma_{b_\ell}^{-1} (\hat{f}_\ell(x_b) - \mu_{b_\ell,c})$
    Score de confiança: $M_\ell = \max_c D'_c$
end for
return Score final de confiança: $\sum_\ell \alpha_\ell M_\ell$

Legenda:

- x — Amostra de teste.
- $\hat{f}(x)$ — Representação normalizada da amostra via função de extração de características.
- $\hat{f}_\ell(x)$ — Representação normalizada da amostra na camada ℓ.
- $\mu_{b_\ell,c}$ — Vetor de média da classe c na camada ℓ.
- Σ_{b_ℓ} — Matriz de covariância estimada na camada ℓ.
- c — Vetor classe usado no treino
- D_c — Distância de Mahalanobis entre $\hat{f}_\ell(x)$ e a classe c.
- $\hat{c}$ — Classe mais próxima de acordo com a distância de Mahalanobis.
- x_b — Amostra perturbada adversarialmente.
- D'_c — Distância de Mahalanobis entre $\hat{f}_\ell(x_b)$ e a classe c.
- M_ℓ — Score de confiança na camada ℓ.
- α_ℓ — Peso atribuído à camada ℓ na combinação final.
- $M(x)$ — Vetor de scores por camada.

3.7 PIPELINE DO NIDS IMPLEMENTADO

Na figura 7 é mostardo o pipeline do modelo implementado no presente trabalho. Os dados tabulares são a entrada do modelo DL treinado adversarialmente por MPB com máscara binária. Seguidamente os dados são analisados pelo detector OOD e inferidos se são in-distribution ou OOD.

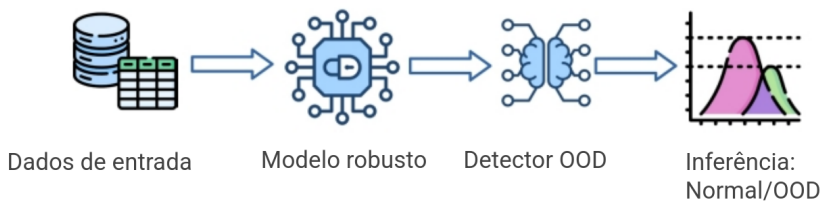


Figura 7 – Dados de entrada > Modelo deep learning treinado adversarialmente > detector OOD > inferência: OOD Limpo ou OOD Adversarial

3.8 MODELO DE AMEAÇA

Os autores em Alatwi e Morisset 2022 definem o modelo de ameaça como um processo sistemático de identificar vulnerabilidades pelo ponto de vista de um atacante, e com isso tomar as medidas de segurança para mitigar essas vulnerabilidades. De acordo com Shostack 2014 esse processo deve identificar:

- **Ativos**: Itens de software ou hardware que atraem o atacante.
- **Superfície de ataque**: Diferentes pontos do sistema que são vulneráveis ao agente malicioso
- **Modelo de adversário**: Características que definem o adversário como motivações e capacidades.
- **Vulnerabilidades** e **Ameaças**: Vulnerabilidades são as fraquezas nos ativos que o agente malicioso pode explorar. Ameaças são eventos onde o atacante explora alguma vulnerabilidade presente nos ativos.
- **Medidas mitigadoras**: Medidas de segurança para prever, detectar ou reduzir o impacto das ameaças.

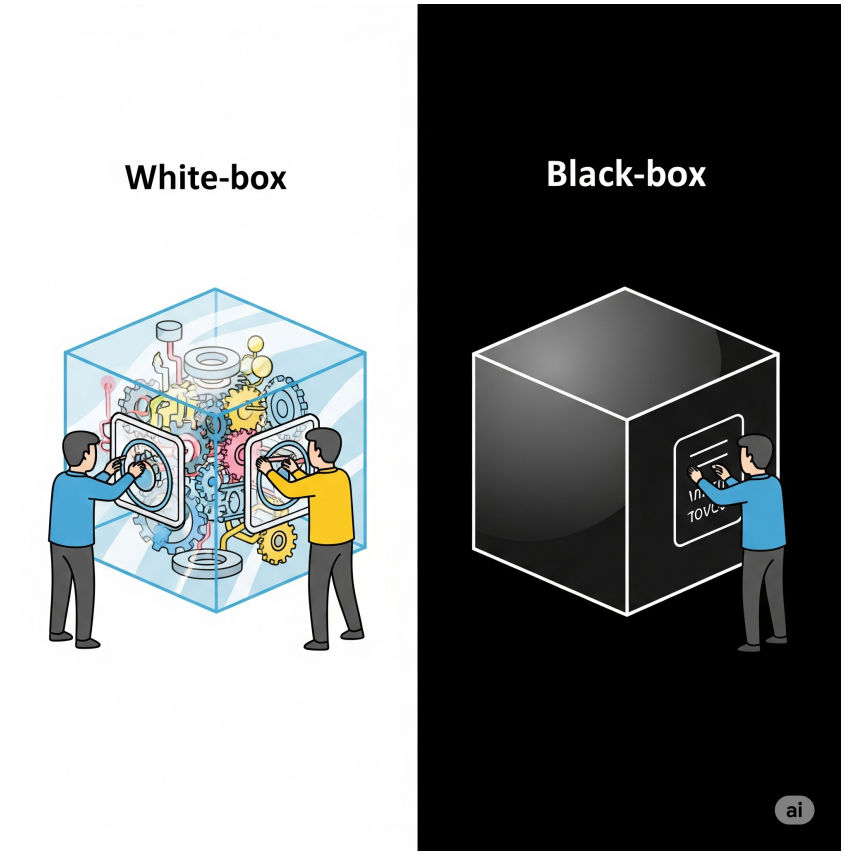
No presente trabalho é identificado:

- Ativos e Superfície de ataque: A rede a ser defendida e o NIDS baseado em DL.
- Vulnerabilidades e Ameaças: Modelos DL por si só são vulneráveis a ataques adversariais, portanto o atacante pode usá-los para evadir o NIDS.
- Modelo de adversário: O adversário pretende atacar a rede e evadir o NIDS utilizando ataques white-box e black-box (Fig 8) ambos não direcionados, ou seja, ataques que poderão se passar por quaisquer uma das classes presentes no dataset de treino.

3.8.1 Ataques adversariais utilizados

Os ataques utilizados são baseados nas normas ℓ_2 e ℓ_∞, com os testes sendo feitos usando a abordagem *white box* e *black box*:

Figura 8 – Modelos de ameça por conhecimento do atacante



White-box se refere a situação onde o atacante consegue acessar o modelo diretamente; Em ataques Black-box o atacante não tem acesso nem conhecimento do modelo alvo

3.8.1.1 White-Box

- Projected Gradient Descent (PGD) Madry et al. 2017: Esse método cria a perturbação a partir do gradiente projetado da função de custo. O faz em vários passos que devem ser ajustados pelo atacante. De acordo com seus autores, também pode ser usado para treinamento adversarial, embora aumente consideravelmente o custo computacional;

- Carlini-Wagner (CW)Carlini e Wagner 2017: Proposto com base na otimização da função:

$$\text{minimize} \quad \|\delta\|_p + c \cdot f(x + \delta)$$

  Onde:

  - $\|\delta\|_p$ é a norma L_p da perturbação δ (por exemplo, L_2 ou L_∞).

- $f(x+\delta)$ é uma função que garante que os dados perturbados sejam classificados incorretamente. Uma das funções propostas é:

$$f(x') = (\max_{i \neq t} Z(x')_i - Z(x')_t)^+$$

onde $Z(x')$ são os logits da rede (as saídas antes da aplicação da função softmax), e $(\cdot)^+$ denota a função ReLU (ou seja, $\max(0, \cdot)$).

- c (*Line Constant*): é uma constante que controla o equilíbrio entre a minimização da perturbação e a garantia de classificação incorreta.

Com este método ,CW, conseguiram gerar a menor perturbação que pode ser baseada nas normas ℓ_0 ,ℓ_2 e ℓ_∞ para criar a amostra evasiva.

3.8.1.2 Black-Box

- Hop Skip Jump Chen e Jordan 2019: Ao invés de usar amostragem, estabelece direção de gradiente a partir de pesquisa binária na fronteira de decisão. Logo após, usa uma busca geométrica para criar a amostra evasiva e por fim realiza uma busca binária de forma que a amostra adversarial não se afaste da fronteira de decisão. É uma versão otimizada do ataque Boundary Brendel, Rauber e Bethge 2017 em relação a quantidade de consultas necessárias para criar uma amostra evasiva. Em geral, consegue criar as amostras com até 10 vezes menos consultas que o ataque Boundary.

- Sign-OPT Cheng et al. 2019 : Estima o sinal da direção do gradiente ao invés do gradiente em si com apenas uma consulta. Em seguida, realiza uma busca para criar a amostra e faz uma busca binária para manter a amostra adversarial próxima a fronteira de decisão. Consegue criar amostras evasivas com 5 a 10 vezes menos consulta se comparado aos dois ataques anteriores.

3.9 AVALIÇÃO DE PERFORMANCE DO CLASSIFICADOR

Na avaliação de modelos de aprendizagem de máquina, é comum o uso de métricas como: Precisão, Recall,ROC-AUC, F1-Score e Acurácia. As definições abaixo foram retiradas da documentação do scikit-learn Pedregosa et al. 2011.

3.9.1 ROC-AUC

Esta métrica indica a capacidade de um modelo em distinguir entre classes positivas e negativas. A Area Under Curve (AUC) (área sob a curva Receiver Operation Characteristic (ROC)) resume o desempenho do modelo em todos os limiares possíveis, variando de 0,5 (desempenho equivalente a um classificador aleatório) a 1,0 (classificação perfeita). Pode ser aplicada a classifcações binárias ou multiclasse do tipo One-vs-Rest e One-vs-One. Por informar a probabilidade de um modelo atribuir pontuação mais alta a uma amostra positiva do que uma negativa, torna a AUC particularmente útil em problemas onde é necessário comparar modelos em diferentes contextos de sensibilidade e especificidade.

Pode ser estimada por:

$$\text{ROC-AUC} = \int_0^1 \text{TPR}(FPR)\, dFPR$$

Onde:

- **True Positive Rate (TPR) (True Positive Rate)**: taxa de verdadeiros positivos, calculada como $TPR = \frac{TP}{TP+FN}$
- **False Positive Rate (FPR) (False Positive Rate)**: taxa de falsos positivos, calculada como $FPR = \frac{FP}{FP+TN}$

3.9.2 Precisão

É a proporção de observações positivas corretamente identificadas para todas as observações positivas previstas. Em outras palavras, a Precisão mede o número de instâncias corretas recuperadas dividido por todas as instâncias recuperadas.

$$\text{Precisão} = \frac{\text{TP}}{(\text{TP} + \text{FP})}$$

Onde:

- **True Positive (TP)**: Total de verdadeiros positivos
- **False Positive (FP)**: Total de falsos positivos

Intuitivamente pode ser ententida com a capacidade do classificador de não rotular como positiva uma amostra que é negativa.

3.9.3 Recall

É a proporção de casos positivos corretamente identificados para todos os casos observados. Em outras palavras, o Recall mede o número de instâncias corretas recuperadas dividido por todas as instâncias corretas. Definido como:

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})}.$$

Onde:

- **TP**: Total de verdadeiros positivos
- **FN**: Total de falsos negativos

O Recall é intuitivamente a habilidade do classificador de encontrar todas as amostras positivas.

3.10 CONTRIBUIÇÕES

No presente trabalho o treino adversarial de Nandi et al. 2023 é adaptado com a máscara binária, de maneira que as perturbações geradas fiquem de acordo com as restrições necessárias para dados tabulares. O detector OOD é adaptado de Lee et al. 2018, entretanto é combinada com a normalização proposta por Müller e Hein 2025, desse modo a combinação dos métodos citados torna possível um detector OOD robusto a amostras adversariais e sem a necessidade que os dados sigam uma distribuição normal. Os ataques adversariais também foram modificados com a máscara binária de maneira que as amostras geradas fossem a partir da restrição causada pela máscara, dessa forma os experimentos com os ataques criam amostras otimizadas para dados tabulares ao invés de usar projeções que tendem a filtrar as amostras evasivas. Os ataques adversários foram tanto white-box quando black-box baseados em consulta e transferência, com isso é avaliada a proposta para cenários onde há uma tentativa de evasão por um agente malicioso interno ou externo.

Deve ser dito que os resultados aqui mostrados para OOD utilizaram uma validação cruzada 10 folds com catboost Dorogush, Ershov e Gulin 2018 e o *grid search* fornecido pela implementação do catboost. O detector OOD dos autores Lee et al. 2018 usa uma validação cruzada com regressão logística, que não se mostrou apropriada para dados desbalanceados e não fornece probabilidades calibradas Bai et al. 2021. Desse modo, após o catboost para score binário do detector OOD, foi realizada uma calibração por classe com predição conforme mondrian Boström, Johansson e Löfström 2021 que é apropriado para dados tabulares, onde a calibração não pode fornecer probabilidades com sobreposição entre as classes. Uma vez que métodos bayesianos sem geral tem alto custo computacional no treinamento e inferência Vonk et al. 2024,Liu et al. 2022,Pape et al. 2023 além do que predição conformal fornece garantias dos conjuntos de predição por classe Boström, Johansson e Löfström 2021 e independem de distribuição, enquanto nos modelos bayesianos a cobertura dos intervalos dependem: das suposições de cada modelo bayesiano e da distribuição a priori, as quais podem ser descalibradas na prática resultando em um modelo DL mal calibrado Portela, Banga e Matabuena 2025, Abdullah, Hassan e Mustafa 2024, Ghosh et al. 2023.

4 EXPERIMENTOS

Este capítulo apresenta os experimentos realizados para avaliar o desempenho da metodologia proposta que referenciamos como TabIDS, frente a ataques adversariais e dados OOD. O objetivo é verificar sua robustez a amostras adversariais, capacidade de generalização e eficácia em comparação à versão original da arquitetura TabNet.

As análises experimentais foram organizadas de forma a avaliar se o treinamento adversarial contribui significativamente para a robustez do modelo em cenários ataque. Ou se o uso de detecção OOD baseada em distância Mahalanobis é eficaz na identificação de amostras anômalas ou adversariais. Investigamos ainda como o modelo se comporta sob diferentes tipos de ataques (white-box e black-box), intensidades de perturbação e normas de distância.

4.1 DATASETS AVALIADOS

4.1.1 UNSW-NB15

Em razão da defasagem de datasets como KDD99 e NSL-KDD, bem como da baixa disponibilidade de conjuntos de dados com características equivalentes e mais atualizados para a época, Moustafa e Slay 2015 criaram o dataset UNSW-NB15. O intuito foi o de amenizar problemas com redundância de dados e representação de ataques mais modernos. O conjunto de dados foi elaborado por meio da ferramenta IXIA PerfectSotorm [1], na Universidade de South Wales (UNSW). Durante o desenvolvimento da proposta, os autores simularam fluxo de rede normal e fluxo associado a ataques, sendo estes capturados pela ferramenta *tcpdump*, uma poderosa ferramenta de linha de comando para análise de pacotes usada para capturar e exibir tráfego de rede. O dataset foi disponibilizado livremente em formato tabular, consistindo em 49 features e vários tipos de ataques: Fuzzers, Dos, Exploits, Backdoors, Worms, Generic, Analysis, Shellcode e Reconnaissance. Contém ao todo 700 mil registros, embora 90% seja de tráfego normal.

[1] https://www.keysight.com/us/en/products/network-test/network-test-hardware/perfectstorm.html

4.1.2 CIC IDS2017

A exemplo do UNSW-NB15, o CIC IDS2017 também foi criado com a preocupação de que os datasets então disponíveis não refletiam as ameaças cibernéticas mais atuais para a sua época. O conjunto de dados foi criado por Sharafaldin et al. 2018 em um ambiente que simula uma rede real com tráfego benigno e malicioso.

O tráfego benigno realista foi gerado com a ferramenta *B-Profile* [2] que simulou vários usuários fazendo requisições em diferentes protocolos.

Os seguintes ataques foram simulados via kali linux: Brute Force, Heartbleed, Botnet, DoS, DDoS, Web Attacks e Infiltration. A simulação resultou em um conjunto de dados rotulado, contendo 80 features extraídos por meio do CICFlowMeter [3]. De acordo com os autores, CIC IDS2017 é superior aos datasets anteriores em termos de diversidade de tráfego, variedade de ataques e características extraídas do fluxo de rede.

4.1.3 CIC IDS2018

Dataset criado pelo Canadian Institute for Cybersecurity 2018 para uso em projetos de pesquisa . É semelhante ao CIC IDS2017 em termos de tipos de ataques e por também usar processamento de arquivos pcap para csv através do CICFlowMeter. A versão csv também é semelhante ao CIC IDS2017 nas features, as quais são compostas por 80 propriedades estatísticas como: comprimento de pacotes, número de pacotes, número de bytes entre outras características que foram estimadas tanto na direção de envio quanto na direção de resposta. A simulação dos ataques se deu em 6 redes com 450 máquinas no total, enquanto no CIC IDS2017 foi coletado em uma única rede de 14 máquinas.

4.2 SETUP EXPERIMENTAL

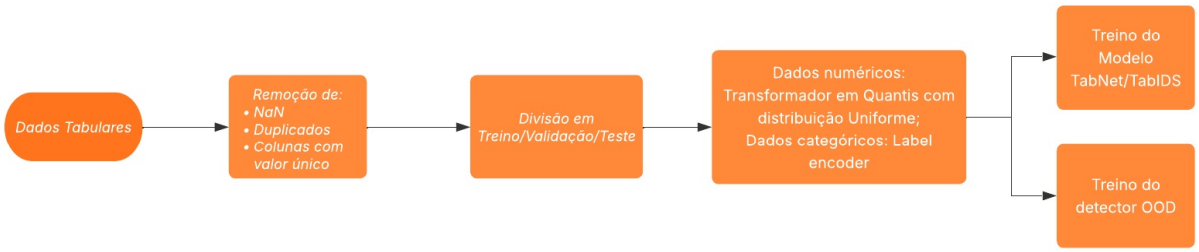
A arquitetura base utilizada é o TabNet. Os dados foram pré-processados da forma: transformador em quantis com distribuição uniforme para dados contínuos, para dados

[2] Encapsula os comportamentos de entidades em uma rede usando diversas técnicas de aprendizado de máquina e análise estatística. As features encapsuladas são distribuições de tamanhos de pacotes de um protocolo, número de pacotes por fluxo, certos padrões na carga útil, tamanho da carga útil e distribuição do tempo de solicitação de um protocolo.

[3] https://www.unb.ca/cic/research/applications.html.

categóricos label encoder (Fig 9). O ajuste de hiperparâmetros se fez em paralelo com uma validação cruzada e otimização com optuna. Deve ser dito que a validação cruzada foi exclusivamente para pesquisa de hiperparâmetros.

Figura 9 – Fluxo do Pré-processamento e treino do modelo



Por decidir usar uma arquitetura que não é comumente associada a NIDS foi realizado um experimento com tempo de inferência com batches = 1 e 10000 repetições para estimar o desvio padrão. Dessa forma pode-se ter a noção da latência média do modelo durante o tráfego de rede. Na Tabela 5 é mostrado que os tempos foram da ordem dos microssegundos (1×10^{-6}), de acordo com Najar e S. 2024,Thorat, Parekh e Mangrulkar 2021 e Cil, Yildiz e Buldu 2021 é um tempo apropriado para um NIDS. Outra avaliação realizada foi comparar os resultados em classificação de anomalia do modelo proposto com outros já feitos na literatura, ver Tabela 6.

Tabela 5 – Tempo de inferência

Dataset	Model	Inference Time (s)		Training Time (h)
		Mean	*StD*	
CIC IDS2017	**TabNet**	2.6×10^{-6}	9.4×10^{-6}	2,5
	TabIDS	8.8×10^{-6}	4.4×10^{-6}	12
UNSW-NB15	**TabNet**	9.9×10^{-6}	1.1×10^{-6}	1
	TabIDS	3.4×10^{-6}	7.7×10^{-6}	10

Para avaliar a robustez do sistema proposto, consideramos diferentes cenários de ataque, abrangendo tanto abordagens white-box quanto black-box, com variações nas normas de perturbação e na intensidade dos ataques. O modelo foi testado com e sem defesa adversarial, permitindo comparações diretas entre a versão original da TabNet e sua variante robusta.

Além disso, foi implementado um mecanismo de detecção de OOD baseado na distância de Mahalanobis, capaz de identificar amostras suspeitas fora da distribuição de

Tabela 6 – Comparativo entre modelos estado-da-arte em NIDS e a TabNet.

	Metric	LSTM	MLP	TabNet
CIC IDS2017	**Precison**	**100** [a]	99.79 [b]	99.84
	Recall	94.44 [a]	99.80 [b]	**99.83**
UNSW-NB15	**Precison**	96.52 [c]	96.7 [d]	**97.16**
	Recall	95.17 [c]	99.80 [d]	**99.66**

Referência: [a] [Dash et al. 2025]; [b] [Cherfi, Lemouari e Boulaiche 2024]; [c] [Ahmed et al. 2025]; [d] [Awotunde, Chakraborty e Adeniyi 2021]; Os valores são para métricas globais.

treinamento. A incorporação de máscaras binárias permitiu adaptar os ataques às restrições semânticas dos dados tabulares, garantindo a validade dos exemplos adversariais e simulando com maior fidelidade cenários realistas de evasão.

4.2.1 Definição do modelo base

Foi realizada uma procura de hiperparâmetros (Tabela 7) por validação cruzada 10 *folds*, com amostras não perturbadas, ou amostras clean, para a TabNet . A divisão dos dados utilizada para treino/validação/teste foi: 60%,20% e 20%, respectivamente. Os dados foram pré-processados com transformador em quantis de distribuição uniforme para mantê-los variando monotonicamente entre 0 e 1.

Após o pré-processamento e treino do modelo, as amostras adversariais foram criadas no conjunto de teste. Primeiro, no TabNet com treino clean, ou seja sem amostras adversariais, e em seguida na TabNet (TabIDS) com treino adversarial.

Nos dados CIC IDS2017, as classes foram mescladas para mitigar os efeitos do desbalanceamento. Numa primeira abordagem:As labels SQL e XSS tornam-se a label Web Attack; As labels FTP e SSH tornam-se a label Brute Force, e as variações de DoS (slowloris, hulk, goldeneye e slowhttptest) tornam-se a label DoS. No entanto, mesmo após esta junção em superclasses (Brute Force, Web Attack e DoS), o problema de desbalanceamento se manteve e portanto este dataset foi avaliado apenas nas classes "Benign",DoS e DDoS, pois foram as que resultaram em melhor desempenho na TabNet. O mesmo problema com desbalanceamento ocorreu com o UNSW-NB15, portanto foram usadas as classes "Normal"e "Reconnaissance"("Recon.") e "Generic", as quais foram melhor reconhecidas tanto no treino normal quanto no treino adversarial. Foram tentados dois métodos

para aliviar os efeitos do desbalanceamento em treino adversarial primeiramente com o UNSW-NB15, entretanto nenhum dos dois obteve um desempenho satisfatório (Anexo C) e no Anexo B estão relacionados os ataques de rede avaliados com o MITRE ATT&CK. Deve ser dito ainda que os modelos: CTGAN Xu et al. 2019, TabDDPM Kotelnikov et al. 2022 e CoDi Lee, Kim e Park 2023, no entanto esses três modelos não geraram bons resultados para os dados minoritários, que nos datasets em questão são os ataques de rede, portanto os experimentos foram conduzidos sem utilizar técnicas para gerar dados sintéticos.

Tabela 7 – Hiperparâmetros da TabNet encontrados pós validação cruzada.

Hiperparâmetro	**CIC IDS2017**		**UNSW-NB15**	
	Binário	***Multiclasse***	***Binário***	***Multiclasse***
WD[a]	42	34	60	28
WA[b]	6	22	43	16
Gamma	1.6	3.4	3.5	1.5
Steps	10	19	7	15
Momentum	0.92	0.73	0.92	0.78

[a] WD: Tamanho de Decisão; [b] WA: Tamanho do *Embedding*.

É importante observar que as features dos conjuntos de dados foram manipuladas conforme descrito em Kuppa et al. 2019, portanto não houve redução de features. Esta abordagem garante que a distinção entre comportamentos benignos e maliciosos seja preservada ao utilizar os ataques adversariais. Para conseguir isso, uma máscara binária é utilizada, permitindo modificar features que podem caracterizar comportamento malicioso, enquanto protege as features que devem permanecer inalteradas.

O modelo TabNet foi treinado usando o otimizador AdamW Loshchilov e Hutter 2017, com uma taxa de aprendizado de 0,002 e a função de perda de entropia cruzada. O desempenho do modelo resultante nas métricas relevantes pode ser conferido nas tabelas 8, para dados clean (não perturbados com amostras adversariais) e nas tabelas 9 para o conjunto com modelos adversariamente treinados. Nas tabelas 10 e 11 estão as métricas resultantes para as classes utilizadas nos experimentos com classificadores multiclasse.

As redes neurais foram testadas contra ataques adversários não direcionados da biblioteca Adversarial Robustness Toolbox Nicolae et al. 2018. Os valores utilizados a seguir para os parâmetros dos ataques white-box e black-box se devem a manter o ataque imperceptível e preservar as restrições necessárias em dados tabulares.

Tabela 8 – Métricas para TabNet em classificação binária e treino clean.

Dataset	Label	Precision	Recall	ROC-AUC
CIC IDS2017	Benign	99.98	99.98	99.98
	Attacks	99.95	99.98	99.96
UNSW-NB15	Normal	99.97	99.97	99.78
	Attacks	97.16	99.62	99.78

Tabela 9 – Métricas para TabIDS com classificação binária.

Dataset	Label	Precision	Recall	ROC-AUC
CIC IDS2017	Benign	99.98	99.98	99.98
	Attacks	99.88	99.98	99.98
UNSW-NB15	Normal	99.98	99.98	99.89
	Attacks	97.21	99.57	99.89

Para ataques white-box, adotamos as seguintes configurações: Ataque PGD com 100 passos e valores epsilon $\epsilon = \{0.1, 0.2, 0.3\}$, onde ϵ é o valor máximo da perturbação no vetor,portanto a pertubação adicionada nas features atacadas somarão como um todo o ϵ selecionado.

O ataque CW usou valores de *Confidence* $c = \{0, 0.2, 0.5\}$ para ambos os conjuntos de dados anteriormente mencionados. O hiperparâmetro Confidence define o quão confiante o modelo-alvo deve estar na classificação errada da amostra adversarial gerada. Valores muito altos são úteis apenas em ataques direcionados. O parâmetro de busca binária é definido como 10. Além disso, a importância relativa da distância e da taxa de aprendizagem foi fixada em $0,01$ para ambos os parâmetros nos conjuntos de dados usados.

Em ataques de black-box, nenhum intervalo de valores foi usado. Para o ataque *HopSkipJump*, foi utilizado um número máximo de avaliações por gradiente igual a 2; ataque *SignOPT* usado para consultas por amostra igual a 200 e um número de direções aleatórias igual a 200.

O método usado para melhorar a robustez foi com o treinamento Adversarial Training with Multiple Perturbation Bounds (AT-MPB) Nandi et al. 2023 com 300 épocas,resultando na TabIDS.

Tabela 10 – Métricas para os modelo multiclasse com CIC IDS2017 e as classes selecionadas.

CIC IDS2017				
Modelo	**Métrica**	**BENIGN**	**DoS**	**DDoS**
TabNet	***Precision***	99.98	99.92	99.97
	Recall	99.98	99.98	99.99
	ROC-AUC	99.99	99.99	99.98
TabIDS	***Precision***	99.99	99.82	99.96
	Recall	99.97	99.96	99.99
	ROC-AUC	99.98	99.97	99.96

Tabela 11 – Métricas para os modelos multiclasse com UNSW-NB15 e as classes selecionadas.

UNSW-NB15				
Modelo	**Métrica**	**Normal**	**Recon.**	**Generic**
TabNet	***Precision***	99.98	96.66	98.48
	Recall	98.85	99.46	99.99
	ROC-AUC	99.43	99.87	99.98
TabIDS	***Precision***	99.99	91.59	96.19
	Recall	99.14	99.26	99.99
	ROC-AUC	99.69	99.93	99.96

4.3 RESULTADOS

Para isso, os experimentos foram conduzidos sobre dois conjuntos de dados amplamente utilizados na literatura: CIC IDS2017 e UNSW-NB15. Foram consideradas classificações binárias e multiclasse, permitindo analisar o desempenho sob diferentes granularidades de labels.

A avaliação abrange ataques adversariais com diferentes estratégias e complexidades, incluindo PGD, CW, HopSkipJump e SignOPT, com variações nos parâmetros de ataque. Cada ataque foi aplicado tanto em modelos treinados com dados clean e em modelos adversarialmente treinados. A performance classificatória foi quantificada por métricas como Precisão, Recall e ROC-AUC, além de métricas específicas para imperceptibilidade das perturbações (Anexo A). No Anexo D estão as figuras referentes a Recall e ROC-AUC, neste capítulo foram usados os gráficos de Precisão pois foi a métrica mais sensível aos ataques adversariais, portanto os ataques utilizados geraram mais falsos positivos que falsos negativos.

Para os ataques PGD e CW, um máximo de 100 e 10 iterações foram usadas respecti-

vamente. Portanto serão representados por PGD-100,CW-10. Para os ataques PGD-100 e HopSkipJump foi usada a norma ℓ_∞. Para os ataques CW-10 e SignoOPT foi usada norma ℓ_2.

4.3.1 Ataque PGD-100

Nos experimentos para modelos de classificação binária utilizando o conjunto de dados CIC IDS2017, após a aplicação do ataque PGD-100 ℓ_∞ com parâmetro $\epsilon = 0,1$, não foi observada redução excessiva na métrica de Precisão (99,36%) para TabNet no label "Attacks", enquanto TabIDS atingiu uma Precisão de 99,94% no mesmo label Tabela 12. Com $\epsilon = 0,3$, os valores de Precisão (87,18%), Recall (97,52%) e ROC-AUC (97,88%) para TabNet foram menores do que aqueles para TabIDS, que relatou Precisão (92,61%), Recall (99,98%) e ROC-AUC (100,00%) para o label "Attacks".

Em modelos multiclasse com CIC IDS2017, observou-se que tanto a Precisão quanto a Recall foram superiores no modelo TabIDS em comparação com o TabNet em vários valores de ϵ. Todos os três labels demonstraram melhorias na métrica ROC-AUC, com melhorias notáveis na Precisão e Recall para os labels “DoS” e “DDoS”, conforme ilustrado na Fig. 10 para a Precisão.

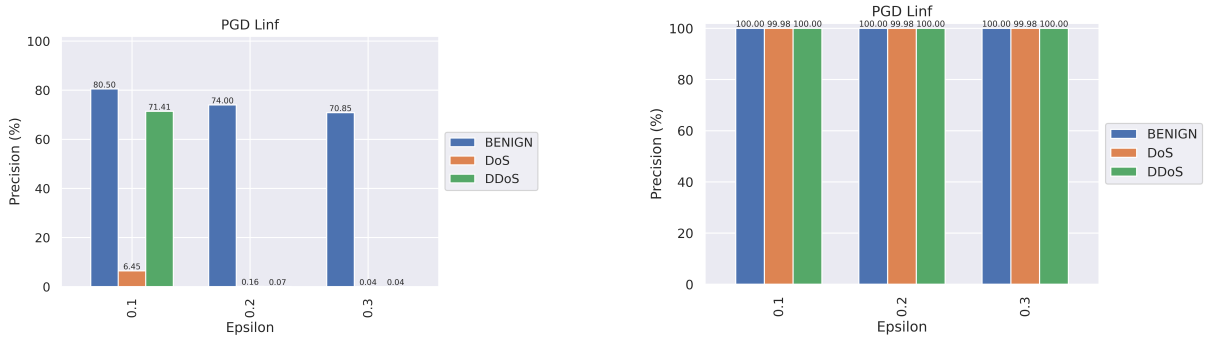
Tabela 12 – Ataque PGD-100 contra classificadores binários com dados CIC IDS2017.

Model	**Metric**	**Epsilon**					
		0.1		***0.2***		***0.3***	
		Benign	**Attacks**	**Benign**	**Attacks**	**Benign**	**Attacks**
TabNet	Precision	99.96	99.36	99.88	94.04	99.17	87.18
	Recall	99.79	99.95	97.96	99.63	95.37	97.52
	ROC-AUC	99.99	99.95	99.69	99.69	97.88	97.88
TabIDS	Precision	99.98	99.94	99.98	99.93	99.97	99.93
	Recall	99.98	99.97	99.97	99.98	99.98	99.97
	ROC-AUC	99.97	99.97	99.98	99.97	99.97	99.96

Tabela 13 – Ataque PGD-100 contra TabNet e TabIDS em classificação multiclasse com dados CIC IDS2017

Models	Metrics	Epsilon								
		0.1			*0.2*			*0.3*		
		Benign	DoS	DDoS	Benign	DoS	DDoS	Benign	DoS	DDoS
TabNet	Precision	80.50	6.45	71.41	74.00	0.16	0.07	70.85	0.04	0.04
	Recall	86.12	3.13	11.70	70.30	0.16	0.03	62.02	0.05	0.02
	ROC-AUC	61.13	42.01	73.11	42.75	20.69	48.66	35.41	15.63	37.19
TabIDS	Precision	99.98	99.82	99.97	99.98	99.82	99.98	99.97	99.81	99.96
	Recall	99.98	99.97	99.98	99.98	99.97	99.98	99.97	99.96	99.97
	ROC-AUC	99.98	99.98	99.98	99.98	99.98	99.98	99.98	99.98	99.98

Figura 10 – Precisão após PGD-100 com CIC IDS2017



Esquerda: TabNet; Direita: TabIDS

Em experimentos na UNSW-NB15 com classificadores binários, para a métrica Precisão no TabNet, o label "Attacks"apresentou valores entre 67% e 68%, enquanto o TabIDS atingiu 90%. Em relação à métrica Recall, o TabIDS demonstrou desempenho marginalmente superior ao TabNet para o label "Normal", com nenhum dos modelos apresentando pontuação abaixo de 99% para essa métrica nos três parâmetros utilizados no ataque PGD Tabela 14.

Os resultados para os modelos multiclasse, Precisão e Recall para o TabIDS, superaram os do TabNet para os labels Recon e Generic para todos os parâmetros do ataque PGD. Fig. 11.

Tabela 14 – Ataque PGD-100 contra TabNet e TabIDS em classificação binária com dados UNSW-NB15.

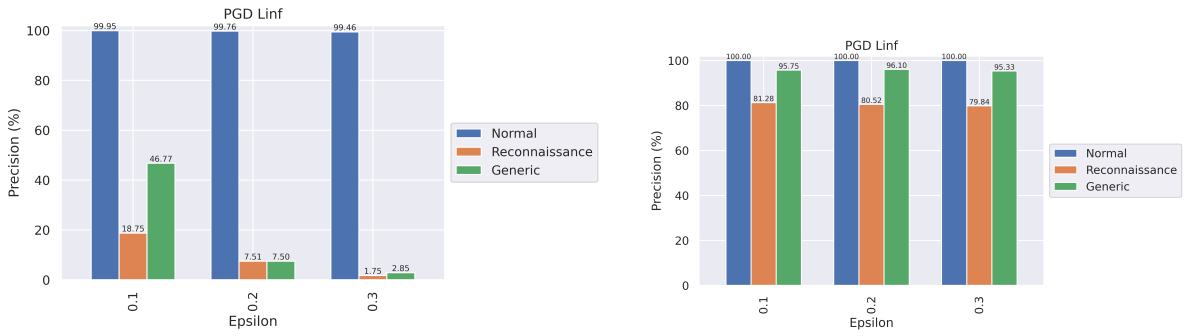
Model	Metric	Epsilon					
		0.1		*0.2*		*0.3*	
		Benign	Attacks	Benign	Attacks	Benign	Attacks
TabNet	Precision	99.88	68.21	99.88	67.80	99.87	67.61
	Recall	99.59	99.89	99.59	99.73	99.59	98.87
	ROC-AUC	99.92	99.92	99.85	99.85	99.82	99.82
TabIDS	Precision	99.90	91.99	99.90	91.67	99.90	90.67
	Recall	99.92	99.68	99.92	99.68	99.91	99.46
	ROC-AUC	99.94	99.94	99.94	99.94	99.94	99.94

Com o ataque PGD-100 ℓ_∞ em classificadores multiclasse se manteve a tendência esperada que as métricas para TabIDS serem maiores para TabNet. O ponto mais baixo para TabIDS é na label Recon. para ϵ=0.3 Fig 11. Embora indique que mesmo após o treinamento adversarial, o desbalanceamento presente no dataset ainda influenciou na desempenho.

Tabela 15 – Ataque PGD-100 contra TabNet e TabIDS em classificação muticlasse com dados UNSW-NB15.

Models	Metrics	Epsilon								
		0.1			*0.2*			*0.3*		
		Normal	**Recon.**	**Generic**	**Normal**	**Recon.**	**Generic**	**Normal**	**Recon.**	**Generic**
TabNet	Precision	99.95	18.75	46.77	99.76	7.51	7.50	99.46	1.75	2.85
	Recall	98.93	10.23	42.35	98.26	5.40	5.52	97.44	1.70	2.46
	ROC-AUC	99.65	98.21	96.74	98.84	95.54	74.90	91.78	93.00	48.46
TabIDS	Precision	99.98	81.28	95.75	99.98	80.52	96.10	99.96	79.84	95.33
	Recall	99.47	90.06	95.81	99.43	88.07	95.08	99.45	85.51	93.75
	ROC-AUC	99.83	98.47	98.89	99.79	98.37	98.62	99.76	98.34	98.34

Figura 11 – Precisão após PGD-100 com UNSW-NB15



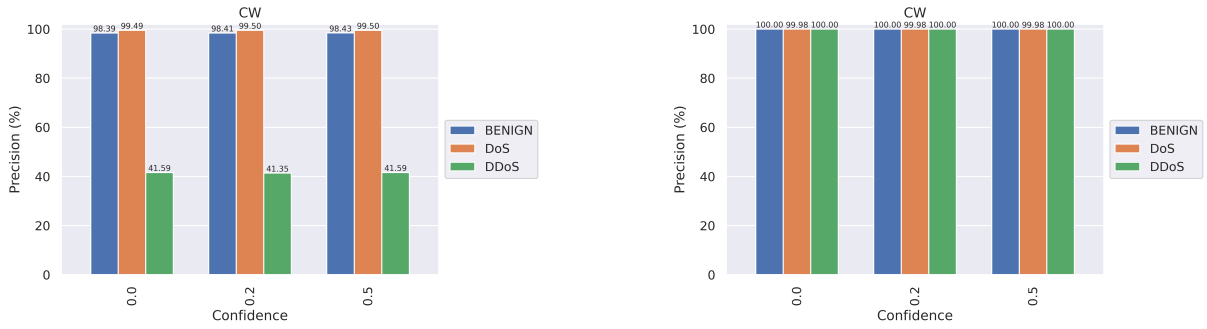
Esquerda: TabNet; Direita: TabIDS

4.3.2 Ataque CW-10

Ao executar ataques CW-10 no CIC IDS2017 com modelos binários, o TabNet apresentou desempenho em torno de 96% a 97% para o label "Benign"e abaixo de 79% para a label "Attacks"na métrica Precisão. Para Recall a label "Attack"ficou entre 69% e 72% . A TabIDS obteve melhores resultados gerais,mantendo a robustez do modelo a esse tipo de ataque para os valores de "Confidence"testados (Tabela 16).

Com modelos multiclasse, o pior desempenho da TabNet foi na label "DDoS". Enquanto as métricas para TabIDS superaram Precision,Recall e ROC-AUC nos três labels, conforme mostrado na Fig 12 e na Tabela 17.

Figura 12 – Ataque CW-10 contra classificadores multiclasse e dados do CIC IDS2017.



Esquerda: TabNet; Direita: TabIDS

Ao conduzir o ataque CW-10 com o conjunto de dados UNSW-NB15 usando classificadores binários, a Precisão foi notavelmente melhor para o label "Attacks"no modelo TabIDS, com "Confidence"por volta dos 92%, enquanto o TabNet caiu para 37,62%. Não foram observadas diferenças relevantes entre TabNet e TabIDS para Recall e ROC-AUC, pois ambos os modelos produziram resultados em torno de 99%. Indicando que nesse caso o método CW-10 gerou mais falsos positivos.

Nos modelos multiclasse, as métricas melhoraram no modelo TabIDS para todos os valores de "Confidence"usados nos experimentos (Fig. 13, que mostra os resultados para Precisão). Recall e ROC-AUC apresentam padrão idêntico.

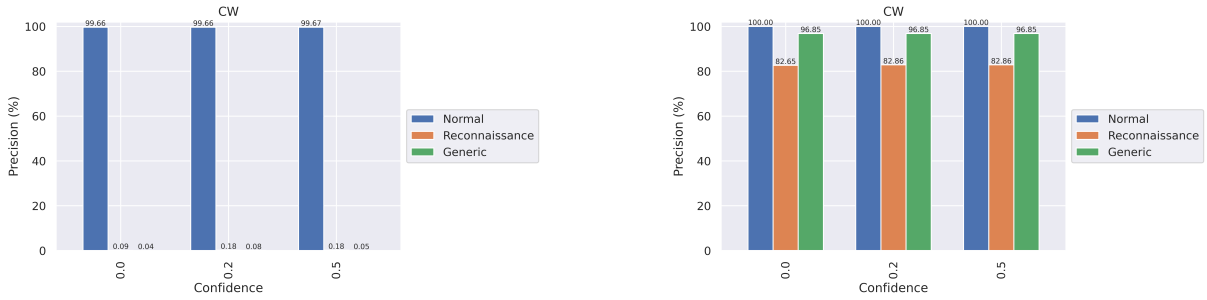
Tabela 16 – Ataque CW-10 contra classificadores binários e dados CIC IDS2017.

Model	Metric	Confidence					
		0.0		*0.2*		*0.5*	
		Benign	Attacks	Benign	Attacks	Benign	Attacks
TabNet	Precision	97.38	78.46	96.21	75.97	96.57	78.12
	Recall	91.16	69.42	91.20	71.54	92.37	72.18
	ROC-AUC	92.24	90.22	93.31	91.48	93.94	92.05
TabIDS	Precision	99.97	99.88	99.96	99.88	99.97	99.88
	Recall	99.97	99.97	99.96	99.97	99.97	99.96
	ROC-AUC	99.97	99.97	99.96	99.97	99.96	99.96

Tabela 17 – Ataques CW-10 contra classificadores multiclasse com CIC IDS2017.

Model	Metric	Confidence								
		0.0			*0.2*			*0.5*		
		Benign	DoS	DDoS	Benign	DoS	DDoS	Benign	DoS	DDoS
TabNet	Precision	98.39	99.49	41.59	98.41	99.50	41.35	98.43	99.50	41.59
	Recall	98.80	90.03	1.98	98.82	90.12	1.96	98.87	90.25	1.98
	ROC-AUC	97.91	93.23	99.63	97.91	93.21	99.62	97.89	93.18	99.61
TabIDS	Precision	99.97	99.96	99.95	99.96	99.96	99.95	99.96	99.95	99.95
	Recall	99.97	99.96	99.96	99.97	99.96	99.95	99.96	99.96	99.95
	ROC-AUC	99.97	99.95	99.95	99.97	99.95	99.95	99.96	99.95	99.95

Figura 13 – Ataque CW-10 contra TabNet and TabIDS em classificação multiclasse e dados do UNSW-NB15.



Esquerda: TabNet; Direita: TabIDS

Ao conduzir o ataque CW-10 com o conjunto de dados UNSW-NB15 usando classificadores binários, as métricas indicaram que o modelo TabIDS melhorou em relação ao modelo TabNet. Notavelmente, a Precisão para a label "Attacks"foi melhor no modelo TabIDS permanecendo em torno de para 92%. Contudo, o Recall para a label "Normal"na TabIDS foi marginalmente maior, mantendo o valor em torno de 99% contra 98% da TabNet Tabela 18.

O valor de ROC-AUC se manteve constante no modelo TabIDS, que atingiu 99,94% em comparação com 99,30% para o modelo TabNet. Nos classificadores multiclasse, ROC-AUC foi maior no modelo TabIDS sobre TabNet para todas as métricas e nos três labels: “Normal”, “Reconnaissance” e “Generic”. Embora a Precisão na TabIDS tenha sido pouco maior que na TabNet para a label “Normal”, Tabela 19, o desempenho da TabIDS foi melhorado nas labels “Recon.” (82%) e "Generic"(96%), porém o modelo TabNet ficou abaixo de 2% para essas mesmas labels. A Recall permaneceu no TabIDS 92% par Recon. e 96% para Generic. Para ROC-AUC em todas as labels e valores de "Confidence"a TabIDS superou a TabNet.

Tabela 18 – Ataque CW-10 contra TabNet and TabIDS em classificação binária e dados do UNSW-NB15.

Model	**Metric**	**Confidence**					
		0.0		***0.2***		***0.5***	
		Benign	**Attacks**	**Benign**	**Attacks**	**Benign**	**Attacks**
TabNet	**Precision**	99.96	37.62	99.96	37.59	99.95	37.56
	Recall	98.56	99.68	98.56	99.68	98.56	99.68
	ROC-AUC	99.30	99.30	99.30	99.30	99.30	99.30
TabIDS	**Precision**	99.96	92.73	99.96	92.59	99.95	92.50
	Recall	99.93	99.68	99.93	99.68	99.93	99.68
	ROC-AUC	99.94	99.94	99.94	99.94	99.94	99.94

Tabela 19 – Ataque CW-10 contra TabNet and TabIDS em classificação multiclasse e dados do UNSW-NB15.

Models	**Metrics**	**Confidence**								
		0.0			***0.2***			***0.5***		
		Normal	**Recon.**	**Generic**	**Normal**	**Recon.**	**Generic**	**Normal**	**Recon.**	**Generic**
TabNet	Precision	99.66	0.09	0.04	99.66	0.18	0.08	99.67	0.18	0.05
	Recall	70.04	0.28	0.27	70.31	0.57	0.47	70.70	0.57	0.33
	ROC-AUC	90.09	97.58	90.39	89.94	97.60	90.21	89.72	97.59	90.04
TabIDS	Precision	99.97	82.65	96.85	99.97	82.86	96.85	99.97	82.86	96.85
	Recall	99.52	92.05	96.01	99.52	92.05	96.01	99.52	92.05	96.01
	ROC-AUC	99.88	98.65	99.17	99.88	98.65	99.17	99.88	98.65	99.17

4.3.3 Hop Skip Jump e Sign-OPT

Os ataques *black-box* aos dados do CIC IDS2017 para classificadores binários, conforme mostrado na Tabela 20, indicam que o modelo TabIDS superou o modelo TabNet em métricas relacionadas a label "Benign". Nos classificadores multiclasse, a Tabela 21 mostra que os valores das métricas para TabIDS são ligeiramente superiores aos do Tab-

Net. No entanto, para SignOPT, todos os valores para TabIDS são um pouco superiores aos do modelo TabNet.

Tabela 20 – Ataques Black-Box contra modelos de classificação binária com dados CIC IDS2017.

Model	Attacks	Metrics	Label	
			Benign	Attacks
TabNet	HopSkipJump	Precision	83.02	16.82
		Recall	67.81	31.93
		ROC-AUC	73.71	73.92
	SignOPT	Precision	85.73	22.51
		Recall	69.88	42.93
		ROC-AUC	80.60	81.02
TabIDS	HopSkipJump	Precision	83.10	17.07
		Recall	83.80	16.36
		ROC-AUC	86.37	86.38
	SignOPT	Precision	86.98	40.52
		Recall	89.82	34.01
		ROC-AUC	93.21	93.22

Tabela 21 – Attacks Black-Box contra modelos de classificação multiclasse com dados CIC IDS2017.

Model	Attacks	Metrics	Label		
			Benign	*DoS*	*DDoS*
TabNet	HopSkipJump	Precision	79.79	13.37	4.71
		Recall	57.59	4.67	0.02
		ROC-AUC	63.25	95.63	99.87
	SignOPT	Precision	81.84	18.73	21.05
		Recall	60.70	11.31	1.16
		ROC-AUC	70.63	93.96	99.22
TabIDS	HopSkipJump	Precision	83.17	10.56	8.27
		Recall	72.13	11.66	3.82
		ROC-AUC	82.96	94.62	98.68
	SignOPT	Precision	86.65	23.13	34.06
		Recall	74.11	19.78	17.69
		ROC-AUC	88.30	96.33	98.83

Nos dados da UNSW-NB15, os classificadores binários mostrados na Tabela 22. Os valores de Precisão e Recall para os ataques HopSkipJump e SignOPT foram ligeiramente maiores no TabIDS dentro dos labels de Ataque.

Nos classificadores multiclasse, Tabela 23, HopSkipJump dentro do TabIDS, a Precisão apresentou resultados ligeiramente melhores para o label "Normal", enquanto a mé-

trica de Recall favoreceu “Reconnaissance”. Além disso, os labels “Generics” resultaram em valores ligeiramente maiores na métrica do que no modelo TabNet.

Tabela 22 – Ataques Black-Box contra modelos de classificação binária com dados UNSW-NB15.

Model	**Attacks**	**Metrics**	**Label**	
			Normal	**Attacks**
TabNet	HopSkipJump	Precision	97.95	1.95
		Recall	89.45	10.06
		ROC-AUC	90.36	90.37
	SignOPT	Precision	98.50	6.71
		Recall	90.01	34.46
		ROC-AUC	93.33	93.36
TabIDS	HopSkipJump	Precision	97.97	2.10
		Recall	83.08	17.38
		ROC-AUC	85.89	85.89
	SignOPT	Precision	98.72	6.85
		Recall	87.00	45.78
		ROC-AUC	92.90	92.90

Tabela 23 – Ataques Black-Box contra modelos de classificação multiclasse com dados UNSW-NB15.

Model	**Attacks**	**Metrics**	**Label**		
			Normal	***Recon.***	***Generic***
TabNet	HopSkipJump	Precision	99.33	6.12	1.47
		Recall	93.26	1.70	0.07
		ROC-AUC	96.19	99.46	99.27
	SignOPT	Precision	99.60	1.80	1.14
		Recall	93.64	1.42	0.07
		ROC-AUC	98.17	99.46	99.09
TabIDS	HopSkipJump	Precision	99.50	6.27	5.07
		Recall	51.87	23.01	0.47
		ROC-AUC	96.08	97.73	97.13
	SignOPT	Precision	99.73	2.01	15.26
		Recall	68.50	16.19	10.24
		ROC-AUC	98.12	97.95	97.69

Tabela 24 – Métricas OOD para TabIDS binária com dados CIC IDS2017.

OOD		
Attacks	**Parameter**	**ROC-AUC**
PGD	0.1	90.5
	0.2	95.8
	0.3	97.2
CW	0	93
	0.2	93.2
	0.5	95.2
Hopskipjump	–	90.7
SignOPT	–	87.7

4.3.4 Detecção de *Out-of-Distribution*

Apresentamos os resultados relacionados ao OOD após ataques adversários contra o modelo TabIDS, incluindo resultados de classificação binária e multiclasse.

Para os modelos de classificação binária e conjuntos de dados avaliados, o método de detecção obteve ROC-AUC superior a 90% para os ataques PGD-100 e CW-10 Tabelas 25, 26 e Fig. 14. Padrões semelhantes são observados nos métodos multiclasse Fig. 15 nos ataques mencionados anteriormente. No entanto, para ataques de black-box, o ROC-AUC para HopSkipJump não excedeu 90% em todos os modelos e conjuntos de dados. Enquanto o SignOPT não ultrapassou 90% em três experimentos, exceto para o TabIDS binário em UNSW-NB15.

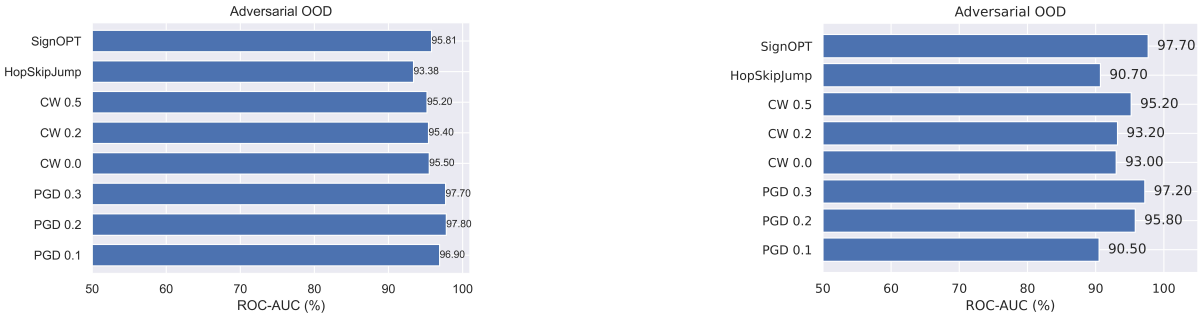
Tabela 25 – Detecção de OOD em classificação binária e dados CIC IDS2017.

OOD		
Attacks	**Parameter**	**ROC-AUC**
PGD	0.1	90.5
	0.2	95.8
	0.3	97.2
CW	0	93
	0.2	93.2
	0.5	95.2
Hopskipjump	–	90.7
SignOPT	–	87.7

Tabela 26 – Detecção de OOD em classificação multiclasse e dados CIC IDS2017.

OOD		
Attack	**Parameter**	**ROC-AUC**
PGD-100	0.1	96.5
	0.2	97.5
	0.3	98.2
CW-10	0	98.7
	0.2	98.4
	0.5	98.1
Hopskipjump	–	80.5
SignOPT	–	86.3

Figura 14 – Detecção OOD com modelo binário



Esquerda: UNSW-NB15; Direita: CIC IDS2017

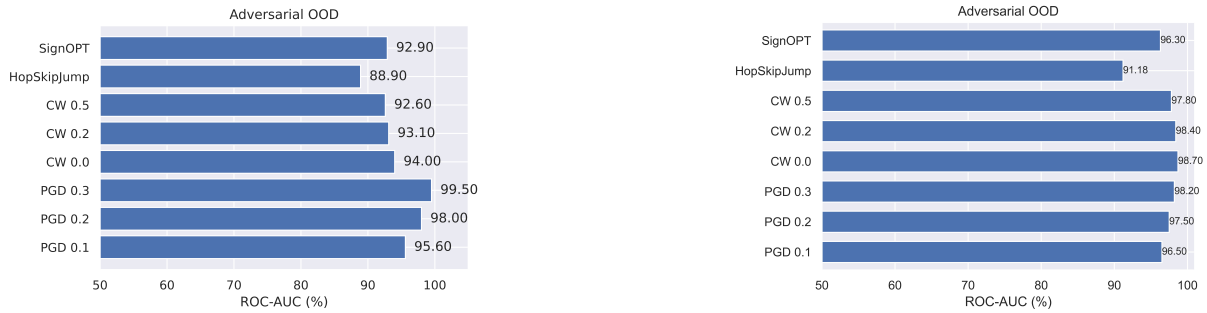
Tabela 27 – Detecção de OOD em classificação binária e dados UNSW-NB15.

OOD		
Attacks	**Parameters**	**ROC-AUC**
PGD-Linf	0.1	96.9
	0.2	97.8
	0.3	97.7
CW	0	95.5
	0.2	95.4
	0.5	95.2
Hopskipjump	–	80
SignOPT	–	82

Tabela 28 – Detecção de OOD em classificação multiclasse e dados UNSW-NB15.

OOD		
Attacks	**Parameters**	**ROC-AUC**
PGD	0.1	95.6
	0.2	98
	0.3	99.5
CW	0	94
	0.2	93.1
	0.5	92.6
Hopskipjump	–	88.9
SignOPT	–	92.9

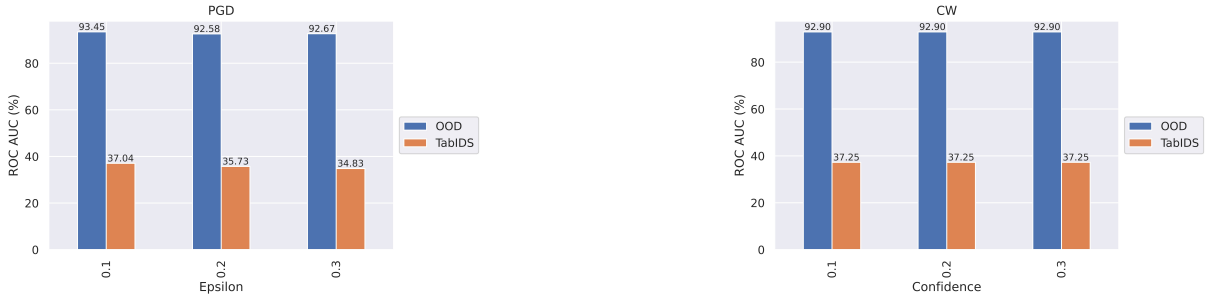
Figura 15 – Detecção OOD com modelo multiclasse.



Esquerda: UNSW-NB15; Direita: CIC IDS2017

4.3.5 OOD para ataques de transferência

Figura 16 – Detecção OOD após transferência dos ataques PGD-100 e CW-10.

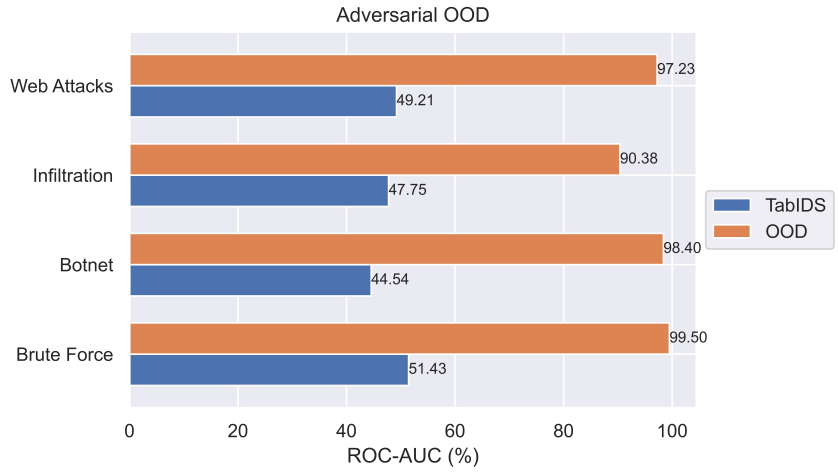


Ataques transferidos a partir de uma MLP para a TabIDS

4.3.6 OOD para dados Clean

Também foi realizado um experimento com ataques CIC IDS 2018 Canadian Institute for Cybersecurity 2018 não observados no treinamento com CIC IDS 2017. Dessa forma, pudemos testar o comportamento do modelo proposto contra ataques de rede não observados no treinamento. Os resultados são mostrados na Figura. 17.

Figura 17 – OOD para dados não perturbados com CIC IDS2018



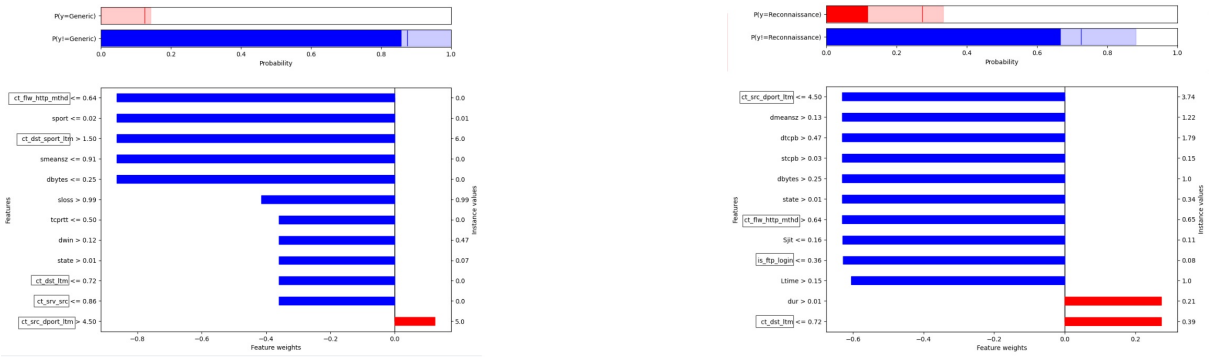
4.4 CONSIDERAÇÕES SOBRE OS RESULTADOS

Em geral, o TabIDS se manteve robusta a ataques adversariais, como evidenciado por suas pontuações de Precisão em cenários de ataques de white-box. Deve-se observar que os valores de Precisão e Recall para a TabIDS foram relativamente equilibrados, sugerindo uma menor suscetibilidade a falsos positivos e falsos negativos em comparação com o TabNet, como também uma melhor identificação dos ataques de rede. Os valores de ROC-AUC para TabIDS evidenciam que o modelo manteve uma boa separabilidade entre as classes. Essa tendência para as métricas não se manteve para experimentos com ataques black-box, embora o detector OOD tenha mitigado o problema e se mostrou útil na detecção de OOD sem perturbações adversariais para ataques de rede presentes no CIC IDS2018, os quais não foram vistos no treino e a detecção OOD também foi útil para detectar ataques por transferência. Deve ser feita a observação que para o ataque de rede *Infiltration* do CIC IDS2018 o valor para ROC-AUC ficou em torno de 90%, apesar de não ser um valor comparável ao estado da arte e o fato que o detector OOD utilizado no

presente trabalho é focado em discriminar amostras adversariais, os trabalhos Hendrycks e Gimpel 2017,Liang, Li e Srikant 2018 sobre detecção OOD permitem concluir que é uma detecção razoável.

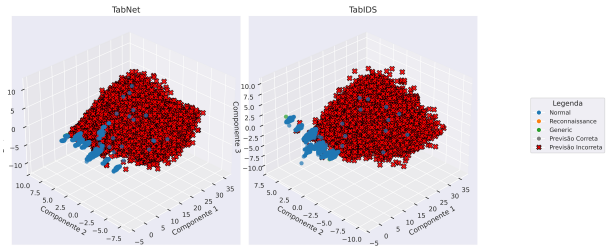
Na figura 18 é mostrada a importância das features após o ataque HopSkipJump contra dados UNSW-NB15. O método utilizado para interpretabilidade é a partir de Lofstrom et al. 2023 que gera interpretações contrafatuais calibradas por predição conforme Papadopoulos et al. 2002. Ainda na figura 18 é possível notar que conforme os ataques fizeram os valores das features (eixo Y) diminuirem, aumentou a chance de erro do classificador, indicando que a combinação entre features não atacáveis e a seleção de features atacáveis contribuiu para uma melhor exploração das vulnerabilidades do classificador TabIDS em relação a fronteira de decisão, ver figura 19, onde é observador para o HopSkipJump, Figura 19 **a**, onde o ataque explorou todo o espaço de ocorrência das amostras e na figura 19 **b**, onde o ataque Sign-OPT explorou melhor a sobreposição das amostras de ataque de rede e fluxo benigno.

Figura 18 – Interpretação por Contrafatual calibrado com ataques de rede do UNSW-NB15.

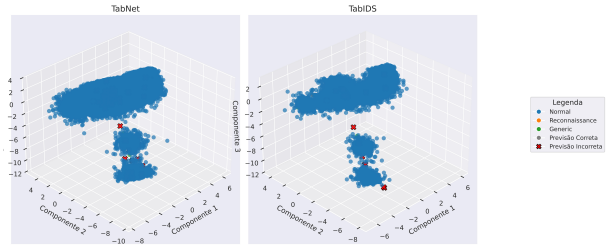


Ataques de rede. Esquerda: Generic; Direita: Reconnaissance

Figura 19 – PCA para fronteira de decisão após ataques Black-Box em dados UNSW-NB15.



a: Ataque HopSkipJump



b: Ataque Sign-OPT

5 CONCLUSÃO E TRABALHOS FUTUROS

A necessidade de proteção das redes em ambientes corporativos se faz urgente. No entanto não é uma tarefa fácil, devido à existência de vários métodos de ataque, surgimento de novas ameças, vulnerabilidades desconhecidas e as formas tradicionais de detecção não conseguem fazer frente ao novos ataques. Além da dificuldade de um lado e novas vulnerabilidades sendo descobertas e exploradas, o quantitativo de dados a ser analisado para confirmar o andamento de um tráfego malicioso é enorme e tende a aumentar, fazendo-se necessárias análises rápidas e que consigam dar conta das novas ameaças.

Este trabalho se propôs a usar deep learning como técnica para detecção de intrusões em redes, uma vez que é uma técnica apropriada para lidar com dados complexos, têm melhor capacidade de generalização e escalabilidade quando comparadas as técnicas tradicionais de ML, sendo portanto melhores na detecção de anomalias em redes de computadores. Para este fim, foi usada a arquitetura TabNet a qual é pensada em dados tabulares, forma de dados comumente usada na publicação dos conjuntos de dados para redes de computadores. No entanto foi levado em consideração que redes neurais são vulneráveis a ataques chamados ataques adversariais, o que para um sistema de cibersegurança aumentaria a chance de ataques e comprometeria a rede a qual deveria defender.Para aumentar a robustez a ataques adversariais evasivos, foi usado treinamento adversarial na TabNet, que no presente trabalho foi chamada de TabIDS, e em seguida um detector de OOD, para eventuais ataques que consigam evadir a TabIDS e ataques de rede não vistos no treinamento. Nos experimentos foram utilizadas pequenas perturbações para testar a eficiência do método TabIDS e compará-lo a uma versão do modelo com treinamento convencional TabNet.

Os efeitos das perturbações foram comparados com base em como elas influenciaram as métricas: Precisão, Recall e ROC-AUC. Nossos resultados mostram que o modelo proposto (TabIDS) teve um bom desempenho no conjunto de dados CIC IDS2017 e UNSW-NB15 para os ataques fortes white-box, ou seja, ataques imperceptíveis mas altamente eficazes em enganar o modelo.O uso de um método OOD ajudou a detectar amostras que o modelo TabIDS não detectou, e essa abordagem foi útil com os ataques black-box. Foi testado ainda o detector OOD em dados não perturbados do CIC IDS2018, para avaliar sua capacidade de detecção em ataques de rede não vistos com o treinamento do CIC

IDS2017. Esses ataques avaliados a partir do CIC IDS2018 resultaram em bons valores com exceção para uma classe ,*Infiltration* que ficou em torno de 90% para ROC-AUC que é considerado razoável pela literatura consultada para OOD.

Uma limitação que deverá ser abordada em trabalhos futuros é a classificação baseada em metadados para fluxo criptografado. Outra limitação a ser explorada é perturbar o fluxo e convertê-lo para pacotes pcap, desse modo observar se os pacotes evadem o modelo, ou até mesmo propor ataques apropriados para dados pcap. Também devem ser investigados ataques gerados por GANs ao invés de adaptar os baseados em visão computacional e fazer a validação do fluxo a oartir da criação e transmissão de pacotes pcap, quantificando tanto o fluxo verdadeiro quanto adversarial produzido por GANs. Além disso,fazem-se necessárias pesquisas sobre quantificação de incertezas que sejam robustas a amostras adversárias e não aumentem a latência do modelo. Essa abordagem aumentaria a confiabilidade dos resultados relacionados à detecção de amostras adversárias, fornecendo intervalos de probabilidade para cada classe de saída do modelo.

REFERÊNCIAS

ABDULGANIYU, O. H.; TCHAKOUCHT, T. A.; SAHEED, Y. K. Towards an efficient model for network intrusion detection system (ids): systematic literature review. *Wireless Networks*, p. 1–30, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:261916036>.

ABDULLAH, A. A.; HASSAN, M. M.; MUSTAFA, Y. T. Leveraging bayesian deep learning and ensemble methods for uncertainty quantification in image classification: A ranking-based approach. *Heliyon*, v. 10, 2024. Disponível em: <https://api.semanticscholar.org/CorpusID:266884358>.

ADHIKARI, D.; JIANG, W.; ZHAN, J.; RAWAT, D. B.; BHATTARAI, A. Recent advances in anomaly detection in internet of things: Status, challenges, and perspectives. *Comput. Sci. Rev.*, v. 54, p. 100665, 2024. Disponível em: <https://api.semanticscholar.org/CorpusID:272096464>.

AHMED, U.; NAZIR, M.; SARWAR, A.; ALI, T.; AGGOUNE, E.-H. M.; SHAHZAD, T.; KHAN, M. A. Signature-based intrusion detection using machine learning and deep learning approaches empowered with fuzzy clustering. *Scientific Reports*, v. 15, n. 1, p. 1726, Jan 2025. ISSN 2045-2322. Disponível em: <https://doi.org/10.1038/s41598-025-85866-7>.

AL, S.; SAğıROğLU Şeref. Explainable artificial intelligence models in intrusion detection systems. *Eng. Appl. Artif. Intell.*, v. 144, p. 110145, 2025. Disponível em: <https://api.semanticscholar.org/CorpusID:276074038>.

ALATWI, H. A.; MORISSET, C. Threat modeling for machine learning-based network intrusion detection systems. *2022 IEEE International Conference on Big Data (Big Data)*, p. 4226–4235, 2022. Disponível em: <https://api.semanticscholar.org/CorpusID:256324917>.

ARIK, S. Ö.; PFISTER, T. Tabnet: Attentive interpretable tabular learning. *ArXiv*, abs/1908.07442, 2019. Disponível em: <https://api.semanticscholar.org/CorpusID:201107047>.

ARJOVSKY, M.; BOTTOU, L.; GULRAJANI, I.; LOPEZ-PAZ, D. Invariant risk minimization. *ArXiv*, abs/1907.02893, 2019. Disponível em: <https://api.semanticscholar.org/CorpusID:195820364>.

ARRECHE, O.; GUNTUR, T.; ROBERTS, J. W.; ABDALLAH, M. E-xai: Evaluating black-box explainable ai frameworks for network intrusion detection. *IEEE Access*, v. 12, p. 23954–23988, 2024. Disponível em: <https://api.semanticscholar.org/CorpusID:267639045>.

AWOTUNDE, J. B.; CHAKRABORTY, C.; ADENIYI, A. E. Intrusion detection in industrial internet of things network-based on deep learning model with rule-based feature selection. *Wireless Communications and Mobile Computing*, v. 2021, n. 1, p. 7154587, 2021. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1155/2021/7154587>.

BAI, Y.; MEI, S.; WANG, H.; XIONG, C. Don't just blame over-parametrization for over-confidence: Theoretical analysis of calibration in binary classification. *Proceedings of the 38th International Conference on Machine Learning, PMLR*, abs/2102.07856, 2021. Disponível em: <https://api.semanticscholar.org/CorpusID:231933881>.

BAJAJ, A.; VISHWAKARMA, D. K. A state-of-the-art review on adversarial machine learning in image classification. *Multimedia Tools and Applications*, v. 83, p. 9351–9416, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:259192171>.

BALAJI, Y.; SANKARANARAYANAN, S.; CHELLAPPA, R. Metareg: Towards domain generalization using meta-regularization. In: *Neural Information Processing Systems*. [s.n.], 2018. Disponível em: <https://api.semanticscholar.org/CorpusID:53979606>.

BISHOP, M. *Computer Security: Art and Science*. 2nd. ed. [S.l.: s.n.], 2018. ISBN 0321712331.

BLANCHARD, G.; LEE, G.; SCOTT, C. D. Generalizing from several related classification tasks to a new unlabeled sample. In: *Neural Information Processing Systems*. [s.n.], 2011. Disponível em: <https://api.semanticscholar.org/CorpusID:15610473>.

BOSTRÖM, H.; JOHANSSON, U.; LÖFSTRÖM, T. Mondrian conformal predictive distributions. In: *International Symposium on Conformal and Probabilistic Prediction with Applications*. [s.n.], 2021. Disponível em: <https://api.semanticscholar.org/CorpusID:243803250>.

BRENDEL, W.; RAUBER, J.; BETHGE, M. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *ArXiv*, abs/1712.04248, 2017. Disponível em: <https://api.semanticscholar.org/CorpusID:2410333>.

BULUSU, S.; KAILKHURA, B.; LI, B.; VARSHNEY, P. K.; SONG, D. Anomalous example detection in deep learning: A survey. *IEEE Access*, IEEE, v. 8, p. 132330–132347, 2020.

Canadian Institute for Cybersecurity. *A Realistic Cyber Defense Dataset (CSE-CIC-IDS2018)*. 2018. Accessed on *04-06-2024* from <https://registry.opendata.aws/cse-cic-ids2018>. Disponível em: <https://registry.opendata.aws/cse-cic-ids2018>.

CAO, K.; WEI, C.; GAIDON, A.; ARÉCHIGA, N.; MA, T. Learning imbalanced datasets with label-distribution-aware margin loss. In: *Neural Information Processing Systems*. [s.n.], 2019. Disponível em: <https://api.semanticscholar.org/CorpusID:189998981>.

CARLINI, N.; WAGNER, D. Towards evaluating the robustness of neural networks. In: IEEE. *2017 ieee symposium on security and privacy (sp)*. [S.l.], 2017. p. 39–57.

CECCARELLI, A.; ZOPPI, T. Intrusion detection without attack knowledge: generating out-of-distribution tabular data. *2023 IEEE 34th International Symposium on Software Reliability Engineering (ISSRE)*, p. 125–136, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:264976983>.

CHAUHAN, R.; SHAH-HEYDARI, S. Polymorphic adversarial ddos attack on ids using gan. *2020 International Symposium on Networks, Computers and Communications (ISNCC)*, p. 1–6, 2020. Disponível em: <https://api.semanticscholar.org/CorpusID:229703720>.

CHEN, J.; JORDAN, M. I. Hopskipjumpattack: A query-efficient decision-based attack. *2020 IEEE Symposium on Security and Privacy (SP)*, p. 1277–1294, 2019. Disponível em: <https://api.semanticscholar.org/CorpusID:173991158>.

CHEN, S.; CARLINI, N.; WAGNER, D. A. Stateful detection of black-box adversarial attacks. *Proceedings of the 1st ACM Workshop on Security and Privacy on Artificial Intelligence*, 2019. Disponível em: <https://api.semanticscholar.org/CorpusID:196470753>.

CHEN, T.; GUESTRIN, C. Xgboost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016. Disponível em: <https://api.semanticscholar.org/CorpusID:4650265>.

CHEN, W.; YANG, K.; YU, Z.; SHI, Y.; CHEN, C. L. P. A survey on imbalanced learning: latest research, applications and future directions. *Artif. Intell. Rev.*, v. 57, p. 137, 2024. Disponível em: <https://api.semanticscholar.org/CorpusID:269668300>.

CHENG, M.; SINGH, S.; CHEN, P. H.; CHEN, P.-Y.; LIU, S.; HSIEH, C.-J. Sign-opt: A query-efficient hard-label adversarial attack. *ArXiv*, abs/1909.10773, 2019. Disponível em: <https://api.semanticscholar.org/CorpusID:202734646>.

CHERFI, S.; LEMOUARI, A.; BOULAICHE, A. Mlp-based intrusion detection for securing iot networks. *Journal of Network and Systems Management*, v. 33, n. 1, p. 20, Dec 2024. ISSN 1573-7705. Disponível em: <https://doi.org/10.1007/s10922-024-09889-7>.

CIL, A. E.; YILDIZ, K.; BULDU, A. Detection of ddos attacks with feed forward based deep neural network model. *Expert Syst. Appl.*, v. 169, p. 114520, 2021. Disponível em: <https://api.semanticscholar.org/CorpusID:232023355>.

CORSINI, A.; YANG, S. J. Are existing out-of-distribution techniques suitable for network intrusion detection? *2023 IEEE Conference on Communications and Network Security (CNS)*, p. 1–9, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:261244119>.

COSTA, J. C.; ROXO, T.; PROENçA, H.; INÁCIO, P. R. M. How deep learning sees the world: A survey on adversarial attacks & defenses. *IEEE Access*, v. 12, p. 61113–61136, 2023.

DAREM, A.; ABAWAJY, J. H.; MAKKAR, A.; ALHASHMI, A. A.; ALANAZI, S. M. Visualization and deep-learning-based malware variant detection using opcode-level features. *Future Gener. Comput. Syst.*, v. 125, p. 314–323, 2021. Disponível em: <https://api.semanticscholar.org/CorpusID:237309473>.

DASH, N.; CHAKRAVARTY, S.; RATH, A. K.; GIRI, N. C.; ABORAS, K. M.; GOWTHAM, N. An optimized lstm-based deep learning model for anomaly network intrusion detection. *Scientific Reports*, v. 15, n. 1, p. 1554, Jan 2025. ISSN 2045-2322. Disponível em: <https://doi.org/10.1038/s41598-025-85248-z>.

DEMONTIS, A.; MELIS, M.; PINTOR, M.; JAGIELSKI, M.; BIGGIO, B.; OPREA, A.; NITA-ROTARU, C.; ROLI, F. Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks. In: *USENIX Security Symposium*. [s.n.], 2018. Disponível em: <https://api.semanticscholar.org/CorpusID:128088823>.

DOROGUSH, A. V.; ERSHOV, V.; GULIN, A. Catboost: gradient boosting with categorical features support. *ArXiv*, abs/1810.11363, 2018. Disponível em: <https://api.semanticscholar.org/CorpusID:26037613>.

DYRMISHI, S.; GHAMIZI, S.; SIMONETTO, T.; TRAON, Y. L.; CORDY, M. On the empirical effectiveness of unrealistic adversarial hardening against realistic adversarial attacks. *2023 IEEE Symposium on Security and Privacy (SP)*, p. 1384–1400, 2022. Disponível em: <https://api.semanticscholar.org/CorpusID:246634982>.

ELSAYED, S.; MOHAMED, K.; MADKOUR, M. A. A comparative study of using deep learning algorithms in network intrusion detection. *IEEE Access*, v. 12, p. 58851–58870, 2024. Disponível em: <https://api.semanticscholar.org/CorpusID:269208272>.

FARAHNAKIAN, F.; HEIKKONEN, J. A deep auto-encoder based approach for intrusion detection system. *2018 20th International Conference on Advanced Communication Technology (ICACT)*, p. 178–183, 2018. Disponível em: <https://api.semanticscholar.org/CorpusID:4335918>.

FARHAN, M.; DIN, H. W. ud; ULLAH, S.; HUSSAIN, M. S.; KHAN, M. A.; MAZHAR, T.; KHATTAK, U. F.; JAGHDAM, I. H. Network-based intrusion detection using deep learning technique. *Scientific Reports*, v. 15, 2025.

GHOSH, S.; BELKHOUJA, T.; YAN, Y.; DOPPA, J. R. Improving uncertainty quantification of deep classifiers via neighborhood conformal prediction: Novel algorithm and theoretical analysis. In: *AAAI Conference on Artificial Intelligence*. [s.n.], 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:257631872>.

GOODFELLOW, I. J.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDE-FARLEY, D.; OZAIR, S.; COURVILLE, A. C.; BENGIO, Y. Generative adversarial networks. *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, p. 1–7, 2021. Disponível em: <https://api.semanticscholar.org/CorpusID:1033682>.

GOODFELLOW, I. J.; SHLENS, J.; SZEGEDY, C. Explaining and harnessing adversarial examples. In: BENGIO, Y.; LECUN, Y. (Ed.). *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. [S.l.: s.n.], 2015.

GRINI, A.; TAHERI, O.; KHAMLICHI, B. E.; SEGHROUCHNI, A. E. F. Constrained network adversarial attacks: Validity, robustness, and transferability. *2025 21st International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*, p. 771–776, 2025. Disponível em: <https://api.semanticscholar.org/CorpusID:278310823>.

HAN, D.; WANG, Z.; ZHONG, Y.; CHEN, W.; YANG, J.; LU, S.; SHI, X.; YIN, X. Evaluating and improving adversarial robustness of machine learning-based network intrusion detectors. *IEEE Journal on Selected Areas in Communications*, v. 39, n. 8, p. 2632–2647, 2021.

HASHEMI, M. J.; KELLER, E. Enhancing robustness against adversarial examples in network intrusion detection systems. In: IEEE. *2020 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. [S.l.], 2020. p. 37–43.

HE, K.; KIM, D. D.; ASGHAR, M. R. Adversarial machine learning for network intrusion detection systems: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, v. 25, p. 538–566, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:255718190>.

HE, Z.; OUYANG, C.; ALZUBAIDI, L.; BARROS, A.; MOREIRA, C. Investigating imperceptibility of adversarial attacks on tabular data: An empirical analysis. *ArXiv*, abs/2407.11463, 2024. Disponível em: <https://api.semanticscholar.org/CorpusID:271218122>.

HENDRYCKS, D.; GIMPEL, K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In: *International Conference on Learning Representations*. [s.n.], 2017. Disponível em: <https://openreview.net/forum?id=Hkg4TI9xl>.

JEBA, S. M.; AURPA, T. T.; ALOM, M. S.; JOY, M. W. M.; AHAMED, A. S.; DASH, S. R.; AHMED, M. S. Recognition of data breach method from story using advanced deep learning. In: *2024 3rd International Conference for Innovation in Technology (INOCON)*. [S.l.: s.n.], 2024. p. 1–6.

JIANG, P.; WU, H.; XIN, C. Deeppose: Detecting gps spoofing attack via deep recurrent neural network. *Digital Communications and Networks*, v. 8, n. 5, p. 791–803, 2022. ISSN 2352-8648. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2352864821000663>.

JMILA, H.; KHEDHER, M. I. Adversarial machine learning for network intrusion detection: A comparative study. *Comput. Networks*, v. 214, p. 109073, 2022. Disponível em: <https://api.semanticscholar.org/CorpusID:249380947>.

KANG, M.-J.; KANG, J.-W. Intrusion detection system using deep neural network for in-vehicle network security. *PLoS ONE*, v. 11, 2016. Disponível em: <https://api.semanticscholar.org/CorpusID:18256723>.

KARUNANAYAKE, N.; GUNAWARDENA, R.; SENEVIRATNE, S.; CHAWLA, S. Out-of-distribution data: An acquaintance of adversarial examples - a survey. Association for Computing Machinery, New York, NY, USA, 2025. ISSN 0360-0300. Disponível em: <https://doi.org/10.1145/3719292>.

KASPERSKY. *O que é uma ameaça persistente avançada (APT)*. 2022. <https://www.kaspersky.com.br/resource-center/definitions/advanced-persistent-threats.>, [Data de acesso: 03-02-2025].

KHOSLA, P.; TETERWAK, P.; WANG, C.; SARNA, A.; TIAN, Y.; ISOLA, P.; MASCHINOT, A.; LIU, C.; KRISHNAN, D. Supervised contrastive learning. *ArXiv*, abs/2004.11362, 2020. Disponível em: <https://api.semanticscholar.org/CorpusID:216080787>.

KIM, S.-H.; WANG, Q.-H.; ULLRICH, J. A comparative study of cyberattacks. *Communications of the ACM*, v. 55, p. 66 – 73, 2012.

KITADA, S.; IYATOMI, H. Making attention mechanisms more robust and interpretable with virtual adversarial training. *Applied Intelligence*, v. 53, p. 15802–15817, 2021. Disponível em: <https://api.semanticscholar.org/CorpusID:233296351>.

KIZZA, J. M. *Guide to Computer Network Security*. 6th. ed. [S.l.]: Springer Publishing Company, Incorporated, 2024. ISBN 3031475488.

KOHLI, M.; CHHABRA, I. A comprehensive survey on techniques, challenges, evaluation metrics and applications of deep learning models for anomaly detection. *Discover Applied Sciences*, 2025. Disponível em: <https://api.semanticscholar.org/CorpusID:280172534>.

KOTELNIKOV, A.; BARANCHUK, D.; RUBACHEV, I.; BABENKO, A. Tabddpm: Modelling tabular data with diffusion models. *ArXiv*, abs/2209.15421, 2022. Disponível em: <https://api.semanticscholar.org/CorpusID:252668788>.

KOULIARIDIS, V.; KAMBOURAKIS, G.; GENEIATAKIS, D. Dissecting contact tracing apps in the android platform. *PLoS ONE*, v. 16, 2020. Disponível em: <https://api.semanticscholar.org/CorpusID:220936590>.

KUPPA, A.; GRZONKOWSKI, S.; ASGHAR, M. R.; LE-KHAC, N.-A. Black box attacks on deep anomaly detectors. In: . New York, NY, USA: Association for Computing Machinery, 2019. ISBN 9781450371643.

LEE, C. E.; KIM, J.; PARK, N. Codi: Co-evolving contrastive diffusion models for mixed-type tabular synthesis. In: *International Conference on Machine Learning*. [s.n.], 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:258309242>.

LEE, K.; LEE, K.; LEE, H.; SHIN, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In: BENGIO, S.; WALLACH, H.; LAROCHELLE, H.; GRAUMAN, K.; CESA-BIANCHI, N.; GARNETT, R. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2018. v. 31. Disponível em: <https://proceedings.neurips.cc/paper_files/paper/2018/file/abdeb6f575ac5c6676b747bca8d09cc2-Paper.pdf>.

LEE, K.; LEE, K.; LEE, H.; SHIN, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, v. 31, 2018.

LI, G.; XU, G.; ZHANG, T. Alleviating the effect of data imbalance on adversarial training. In: . [s.n.], 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:259991727>.

LI, H.; PAN, S. J.; WANG, S.; KOT, A. C. Domain generalization with adversarial feature learning. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, p. 5400–5409, 2018. Disponível em: <https://api.semanticscholar.org/CorpusID:52833113>.

LI, L.; HUANG, C.; CHEN, J. Automated discovery and mapping att&ck tactics and techniques for unstructured cyber threat intelligence. *Comput. Secur.*, v. 140, p. 103815, 2024. Disponível em: <https://api.semanticscholar.org/CorpusID:268581021>.

LIANG, S.; LI, Y.; SRIKANT, R. Enhancing the reliability of out-of-distribution image detection in neural networks. In: *International Conference on Learning Representations*. [s.n.], 2018. Disponível em: <https://openreview.net/forum?id=H1VGkIxRZ>.

LIANG, W.; HU, Y.; ZHOU, X.; PAN, Y.; WANG, K. I.-K. Variational few-shot learning for microservice-oriented intrusion detection in distributed industrial iot. *IEEE Transactions on Industrial Informatics*, v. 18, p. 5087–5095, 2022. Disponível em: <https://api.semanticscholar.org/CorpusID:244334256>.

LIAO, H.; MURAH, M. Z.; HASAN, M. K.; AMAN, A. H. M.; FANG, J.; HU, X.; KHAN, A. U. R. A survey of deep learning technologies for intrusion detection in internet of things. *IEEE Access*, v. 12, p. 4745–4761, 2024. Disponível em: <https://api.semanticscholar.org/CorpusID:266805534>.

LIN, Z.; SHI, Y.; XUE, Z. Idsgan: Generative adversarial networks for attack generation against intrusion detection. In: _____. *Advances in Knowledge Discovery and Data Mining*. Springer International Publishing, 2022. p. 79–91. ISBN 9783031059810. Disponível em: <http://dx.doi.org/10.1007/978-3-031-05981-0_7>.

LIU, J. Z.; PADHY, S.; REN, J. J.; LIN, Z.; WEN, Y.; JERFEL, G.; NADO, Z.; SNOEK, J.; TRAN, D.; LAKSHMINARAYANAN, B. A simple approach to improve single-model deep uncertainty via distance-awareness. *Journal of Machine Learning Research 23*, abs/2205.00403, 2022.

LIU, S.; SUN, J.; LI, J. Query-efficient hard-label black-box attacks using biased sampling. *2020 Chinese Automation Congress (CAC)*, p. 3872–3877, 2020. Disponível em: <https://api.semanticscholar.org/CorpusID:231737079>.

LOFSTROM, H.; LOFSTROM, T.; JOHANSSON, U.; SONSTROD, C. Calibrated explanations: with uncertainty information and counterfactuals. *ArXiv*, abs/2305.02305, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:258461042>.

LONG, M.; CAO, Y.; WANG, J.; JORDAN, M. I. Learning transferable features with deep adaptation networks. *ArXiv*, abs/1502.02791, 2015. Disponível em: <https://api.semanticscholar.org/CorpusID:556999>.

LOSHCHILOV, I.; HUTTER, F. Decoupled weight decay regularization. In: *International Conference on Learning Representations*. [s.n.], 2017. Disponível em: <https://api.semanticscholar.org/CorpusID:53592270>.

LU, Z.; YANG, Y.; ZHU, X.; LIU, C.; SONG, Y.-Z.; XIANG, T. Stochastic classifiers for unsupervised domain adaptation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, p. 9108–9117, 2020. Disponível em: <https://api.semanticscholar.org/CorpusID:219630841>.

MADRY, A.; MAKELOV, A.; SCHMIDT, L.; TSIPRAS, D.; VLADU, A. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

MAHDAVIFAR, S.; GHORBANI, A. A. Application of deep learning to cybersecurity: A survey. *Neurocomputing*, v. 347, p. 149–176, 2019.

MALININ, A.; GALES, M. Reverse kl-divergence training of prior networks: Improved uncertainty and adversarial robustness. *Advances in Neural Information Processing Systems*, v. 32, 2019.

MALININ, A.; GALES, M. J. F. Reverse kl-divergence training of prior networks: Improved uncertainty and adversarial robustness. In: *Neural Information Processing Systems*. [s.n.], 2019. Disponível em: <https://api.semanticscholar.org/CorpusID:173188734>.

MARINHO, R.; HOLANDA, R. Automated emerging cyber threat identification and profiling based on natural language processing. *IEEE Access*, v. 11, p. 58915–58936, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:257678882>.

MASDARI, M.; KHEZRI, H. A survey and taxonomy of the fuzzy signature-based intrusion detection systems. *Appl. Soft Comput.*, v. 92, p. 106301, 2020. Disponível em: <https://api.semanticscholar.org/CorpusID:218821448>.

MASEER, Z. K.; YUSOF, R.; BAHAMAN, N.; MOSTAFA, S. A.; FOOZY, C. F. M. Benchmarking of machine learning for anomaly based intrusion detection systems in the cicids2017 dataset. *IEEE Access*, v. 9, p. 22351–22370, 2021. Disponível em: <https://api.semanticscholar.org/CorpusID:231914286>.

MATHOV, Y.; LEVY, E.; KATZIR, Z.; SHABTAI, A.; ELOVICI, Y. Not all datasets are born equal: On heterogeneous tabular data and adversarial examples. *Knowl. Based Syst.*, v. 242, p. 108377, 2022.

MITRE Corporation. *MITRE ATT&CK®: Adversary Tactics and Techniques*. 2015–2025. <https://attack.mitre.org/>. Accessed: 2 de julho de 2025.

MOOSAVI-DEZFOOLI, S.-M.; FAWZI, A.; FAWZI, O.; FROSSARD, P. Universal adversarial perturbations. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, p. 86–94, 2016. Disponível em: <https://api.semanticscholar.org/CorpusID:11558223>.

MOUSTAFA, N.; SLAY, J. Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set). *2015 Military Communications and Information Systems Conference (MilCIS)*, p. 1–6, 2015. Disponível em: <https://api.semanticscholar.org/CorpusID:18965349>.

MUELLER, M.; HEIN, M. Mahalanobis++: Improving ood detection via feature normalization. *ArXiv*, abs/2505.18032, 2025. Disponível em: <https://api.semanticscholar.org/CorpusID:278886091>.

MUHAMMAD, A.; BAE, S.-H. A survey on efficient methods for adversarial robustness. *IEEE Access*, v. 10, p. 118815–118830, 2022.

MÜLLER, M.; HEIN, M. Mahalanobis++: Improving OOD detection via feature normalization. In: *Forty-second International Conference on Machine Learning*. [s.n.], 2025. Disponível em: <https://openreview.net/forum?id=vutMcZl50l>.

NAJAR, A. A.; S., M. N. A robust ddos intrusion detection system using convolutional neural network. *Comput. Electr. Eng.*, v. 117, p. 109277, 2024. Disponível em: <https://api.semanticscholar.org/CorpusID:269743947>.

NANDI, S.; ADDEPALLI, S.; RANGWANI, H.; BABU, R. V. Certified adversarial robustness within multiple perturbation bounds. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2023. p. 2297–2304.

NICOLAE, M.-I.; SINN, M.; TRAN, M.-N.; BUESSER, B.; RAWAT, A.; WISTUBA, M.; ZANTEDESCHI, V.; BARACALDO, N.; CHEN, B.; LUDWIG, H.; MOLLOY, I.; EDWARDS, B. Adversarial robustness toolbox v1.0.0. *arXiv: Learning*, 2018. Disponível em: <https://api.semanticscholar.org/CorpusID:219890559>.

NOACK, A.; AHERN, I.; DOU, D.; LI, B. A. An empirical study on the relation between network interpretability and adversarial robustness. *SN Computer Science*, v. 2, 2020. Disponível em: <https://api.semanticscholar.org/CorpusID:227311023>.

PAPADOPOULOS, H.; PROEDROU, K.; VOVK, V.; GAMMERMAN, A. Inductive confidence machines for regression. In: *European Conference on Machine Learning*. [s.n.], 2002. Disponível em: <https://api.semanticscholar.org/CorpusID:42084298>.

PAPE, D.; DäUBENER, S.; EISENHOFER, T.; CINà, A. E.; SCHöNHERR, L. On the limitations of model stealing with uncertainty quantification models. In: *ESANN*. [s.n.], 2023. Disponível em: <https://doi.org/10.14428/esann/2023.ES2023-125>.

PAWLICKI, M.; CHORAŚ, M.; KOZIK, R. Defending network intrusion detection systems against adversarial evasion attacks. *Future Generation Computer Systems*, Elsevier, v. 110, p. 148–154, 2020.

PAYA, A.; ARRONI, S.; GARCÍA-DÍAZ, V.; GÓMEZ, A. G. Apollon: A robust defense system against adversarial machine learning attacks in intrusion detection systems. *Comput. Secur.*, v. 136, p. 103546, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:264395027>.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V.; VANDERPLAS, J.; PASSOS, A.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M.; DUCHESNAY, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011.

PENG, Y.; FU, G.; LUO, Y.; HU, J.; LI, B.; YAN, Q. Detecting adversarial examples for network intrusion detection system with gan. In: IEEE. *2020 IEEE 11th International Conference on Software Engineering and Service Science (ICSESS)*. [S.l.], 2020. p. 6–10.

PORTELA, A.; BANGA, J. R.; MATABUENA, M. Conformal prediction for uncertainty quantification in dynamic biological systems. *PLOS Computational Biology*, v. 21, 2025. Disponível em: <https://api.semanticscholar.org/CorpusID:278530675>.

POTLURI, S.; DIEDRICH, C. Accelerated deep neural networks for enhanced intrusion detection system. *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*, p. 1–8, 2016. Disponível em: <https://api.semanticscholar.org/CorpusID:18959456>.

QIAN, Z.; HUANG, K.; WANG, Q.; ZHANG, X.-Y. A survey of robust adversarial training in pattern recognition: Fundamental, theory, and methodologies. *Pattern Recognit.*, v. 131, p. 108889, 2022.

SAITO, K.; WATANABE, K.; USHIKU, Y.; HARADA, T. Maximum classifier discrepancy for unsupervised domain adaptation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, p. 3723–3732, 2017. Disponível em: <https://api.semanticscholar.org/CorpusID:4619542>.

SEHWAG, V.; BHAGOJI, A. N.; SONG, L.; SITAWARIN, C.; CULLINA, D.; CHIANG, M.; MITTAL, P. Analyzing the robustness of open-world machine learning. In: *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*. [S.l.: s.n.], 2019. p. 105–116.

SHARAFALDIN, I.; LASHKARI, A. H.; GHORBANI, A. A. et al. Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp*, v. 1, p. 108–116, 2018.

SHOSTACK, A. Threat modeling: Designing for security. In: . [s.n.], 2014. Disponível em: <https://api.semanticscholar.org/CorpusID:107169324>.

SHU, D.; LESLIE, N. O.; KAMHOUA, C. A.; TUCKER, C. S. Generative adversarial attacks against intrusion detection systems using active learning. In: *Proceedings of the 2nd ACM Workshop on Wireless Security and Machine Learning*. New York, NY, USA: Association for Computing Machinery, 2020. p. 1–6. Disponível em: <https://doi.org/10.1145/3395352.3402618>.

SI, J. Y. H.; COOPER, M.; CHENG, W. Y.; KRISHNAN, R. Interpretabnet: Enhancing interpretability of tabular data using deep generative models and large language models. In: *NeurIPS 2023 Second Table Representation Learning Workshop*. [s.n.], 2023. Disponível em: <https://openreview.net/forum?id=kzR5Cj5blw>.

SINHA, A.; NAMKOONG, H.; DUCHI, J. Certifiable distributional robustness with principled adversarial training. In: *International Conference on Learning Representations*. [s.n.], 2018. Disponível em: <https://openreview.net/forum?id=Hk6kPgZA->.

STALLINGS, W.; BROWN, L. *Computer Security: Principles and Practice*. 4th. ed. USA: Prentice Hall Press, 2018. ISBN 1292220619.

STROM, B. E.; APPLEBAUM, A.; MILLER, D.; NICKELS, K. C.; PENNINGTON, A. G.; THOMAS, C. Mitre att&ck ® : Design and philosophy. In: . [s.n.], 2020. Disponível em: <https://api.semanticscholar.org/CorpusID:251834854>.

SZEGEDY, C.; ZAREMBA, W.; SUTSKEVER, I.; BRUNA, J.; ERHAN, D.; GOODFELLOW, I. J.; FERGUS, R. Intriguing properties of neural networks. In: BENGIO, Y.; LECUN, Y. (Ed.). *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. [s.n.], 2014. Disponível em: <http://arxiv.org/abs/1312.6199>.

TAHERI, S.; KHORMALI, A.; SALEM, M. A.; YUAN, J.-S. Developing a robust defensive system against adversarial examples using generative adversarial networks. *Big Data Cogn. Comput.*, v. 4, p. 11, 2020.

TANG, T. A.; MHAMDI, L.; MCLERNON, D. C.; ZAIDI, S. A. R.; GHOGHO, M. Deep learning approach for network intrusion detection in software defined networking. *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, p. 258–263, 2016. Disponível em: <https://api.semanticscholar.org/CorpusID:15987982>.

TAVALLAEE, M.; BAGHERI, E.; LU, W.; GHORBANI, A. A. A detailed analysis of the kdd cup 99 data set. In: *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*. [S.l.: s.n.], 2009. p. 1–6.

TEAM, T. pandas development. *pandas-dev/pandas: Pandas*. Zenodo, 2020. Disponível em: <https://doi.org/10.5281/zenodo.3509134>.

THAKKAR, A.; LOHIYA, R. A review on machine learning and deep learning perspectives of ids for iot: Recent updates, security issues, and challenges. *Archives of Computational Methods in Engineering*, v. 28, p. 3211 – 3243, 2020.

THORAT, O.; PAREKH, N.; MANGRULKAR, R. S. Taxodacml: Taxonomy based divide and conquer using machine learning approach for ddos attack classification. *Int. J. Inf. Manag. Data Insights*, v. 1, p. 100048, 2021. Disponível em: <https://api.semanticscholar.org/CorpusID:252356601>.

TSIPRAS, D.; SANTURKAR, S.; ENGSTROM, L.; TURNER, A.; MADRY, A. Robustness may be at odds with accuracy. In: *International Conference on Learning Representations*. [s.n.], 2019. Disponível em: <https://openreview.net/forum?id=SyxAb30cY7>.

UMA, M.; PADMAVATHI, G. A survey on various cyber attacks and their classification. *Int. J. Netw. Secur.*, v. 15, p. 390–396, 2013. Disponível em: <https://api.semanticscholar.org/CorpusID:12028088>.

VACCA, J. R. *Network and System Security, Second Edition*. 2nd. ed. [S.l.]: Syngress Publishing, 2013. ISBN 012416689X.

VONK, M. C.; BRAND, S.; MALEKOVIC, N.; BÄCK, T.; LAARMAN, A.; KONONOVA, A. V. Balancing computational cost and accuracy in inference of continuous bayesian networks. In: *European Workshop on Probabilistic Graphical Models*. [S.l.: s.n.], 2024.

WANG, C.; ZHANG, M.; ZHAO, J.; KUANG, X. Black-box adversarial attacks on deep neural networks: A survey. In: IEEE. *2022 4th International Conference on Data Intelligence and Security (ICDIS)*. [S.l.], 2022. p. 88–93.

WANG, N.; CHEN, Y.; XIAO, Y.; HU, Y.; LOU, W.; HOU, Y. T. Manda: On adversarial example detection for network intrusion detection system. *IEEE Transactions on Dependable and Secure Computing*, IEEE, v. 20, n. 2, p. 1139–1153, 2022.

WANG, Q.; LIU, F.; HAN, B.; LIU, T.; GONG, C.; NIU, G.; ZHOU, M.; SUGIYAMA, M. Probabilistic margins for instance reweighting in adversarial training. In: *Neural Information Processing Systems*. [s.n.], 2021. Disponível em: <https://api.semanticscholar.org/CorpusID:235436104>.

WANG, Q.; WANG, Y.; ZHU, H.; WANG, Y. Improving out-of-distribution generalization by adversarial training with structured priors. *Advances in Neural Information Processing Systems*, v. 35, p. 27140–27152, 2022.

WANG, W.; XU, H.; LIU, X.; LI, Y.; THURAISINGHAM, B. M.; TANG, J. Imbalanced adversarial training with reweighting. *2022 IEEE International Conference on Data Mining (ICDM)*, p. 1209–1214, 2021. Disponível em: <https://api.semanticscholar.org/CorpusID:236493567>.

WANG, Y.; ZOU, D.; YI, J.; BAILEY, J.; MA, X.; GU, Q. Improving adversarial robustness requires revisiting misclassified examples. In: *International conference on learning representations*. [S.l.: s.n.], 2019.

WANG, Y.; ZOU, D.; YI, J.; BAILEY, J.; MA, X.; GU, Q. Improving adversarial robustness requires revisiting misclassified examples. In: *International Conference on Learning Representations*. [s.n.], 2020. Disponível em: <https://api.semanticscholar.org/CorpusID:211548864>.

WONG, J. A.; BERENBEIM, A. M.; BIERBRAUER, D. A.; BASTIAN, N. D. Uncertainty-quantified, robust deep learning for network intrusion detection. *2023 Winter Simulation Conference (WSC)*, p. 2470–2481, 2023. Disponível em: <https://api.semanticscholar.org/CorpusID:267339145>.

WU, D.; FANG, B.; WANG, J.; LIU, Q.; CUI, X. Evading machine learning botnet detection models via deep reinforcement learning. In: *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*. [S.l.: s.n.], 2019. p. 1–6.

WU, K.; WANG, A.; YU, Y. Stronger and faster wasserstein adversarial attacks. In: *International Conference on Machine Learning*. [s.n.], 2020. Disponível em: <https://api.semanticscholar.org/CorpusID:221082497>.

XIN, S.; WANG, Y.; SU, J.; WANG, Y. On the connection between invariant learning and adversarial training for out-of-distribution generalization. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. [S.l.: s.n.], 2023. v. 37, n. 9, p. 10519–10527.

XU, H.; MA, Y.; LIU, H.; DEB, D.; LIU, H.; TANG, J.; JAIN, A. K. Adversarial attacks and defenses in images, graphs and text: A review. *International Journal of Automation and Computing*, v. 17, p. 151 – 178, 2019. Disponível em: <https://api.semanticscholar.org/CorpusID:202660800>.

XU, L.; SKOULARIDOU, M.; CUESTA-INFANTE, A.; VEERAMACHANENI, K. Modeling tabular data using conditional gan. In: *Advances in Neural Information Processing Systems*. [S.l.: s.n.], 2019.

YANG, J.; ZHOU, K.; LI, Y.; LIU, Z. Generalized out-of-distribution detection: A survey. *Int J Comput Vis*, v. 132, 2024. Disponível em: <https://doi.org/10.1007/s11263-024-02117-4>.

YANG, Z.; LIU, X.; LI, T.; WU, D.; WANG, J.; ZHAO, Y.; HAN, H. A systematic literature review of methods and datasets for anomaly-based network intrusion detection. *Comput. Secur.*, v. 116, p. 102675, 2022. Disponível em: <https://api.semanticscholar.org/CorpusID:247215059>.

YI, T.; CHEN, X.; ZHU, Y.; GE, W.; HAN, Z. Review on the application of deep learning in network attack detection. *J. Netw. Comput. Appl.*, v. 212, p. 103580, 2022.

YIN, J.; TANG, M.; CAO, J.; WANG, H. Apply transfer learning to cybersecurity: Predicting exploitability of vulnerabilities by description. *Knowledge-Based Systems*, v. 210, p. 106529, 2020. ISSN 0950-7051. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950705120306584>.

YIN, J.; TANG, M.; CAO, J.; YOU, M.; WANG, H.; ALAZAB, M. Knowledge-driven cybersecurity intelligence: Software vulnerability coexploitation behavior discovery. *IEEE Transactions on Industrial Informatics*, v. 19, p. 5593–5601, 2023.

YOU, M.; YIN, J.; WANG, H.; CAO, J.; WANG, K. N.; MIAO, Y.; BERTINO, E. A knowledge graph empowered online learning framework for access control decision-making. *World Wide Web*, v. 26, p. 827–848, 2022. Disponível em: <https://api.semanticscholar.org/CorpusID:250007362>.

YUE, X.; ZHANG, Y.; ZHAO, S.; SANGIOVANNI-VINCENTELLI, A. L.; KEUTZER, K.; GONG, B. Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, p. 2100–2110, 2019. Disponível em: <https://api.semanticscholar.org/CorpusID:202540251>.

ZHANG, C.; COSTA-PÉREZ, X.; PATRAS, P. Tiki-taka: Attacking and defending deep learning-based intrusion detection systems. In: *Proceedings of the 2020 ACM SIGSAC Conference on Cloud Computing Security Workshop*. [S.l.: s.n.], 2020. p. 27–39.

ZHANG, C.; JIA, D. qiu; WANG, L.; WANG, W.; LIU, F.; YANG, A. Comparative research on network intrusion detection methods based on machine learning. *Comput. Secur.*, v. 121, p. 102861, 2022. Disponível em: <https://api.semanticscholar.org/CorpusID:251171549>.

ZHANG, H.; YU, Y.; JIAO, J.; XING, E. P.; GHAOUI, L. E.; JORDAN, M. I. Theoretically principled trade-off between robustness and accuracy. In: CHAUDHURI, K.; SALAKHUTDINOV, R. (Ed.). *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. [S.l.]: PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 7472–7482.

ZHAO, Q.; CHEN, M.; GU, Z.; LUAN, S.; ZENG, H.; CHAKRABORTY, S. Can bus intrusion detection based on auxiliary classifier gan and out-of-distribution detection. *ACM Transactions on Embedded Computing Systems (TECS)*, v. 21, p. 1 – 30, 2022. Disponível em: <https://api.semanticscholar.org/CorpusID:249282075>.

ZHOU, K.; LIU, Z.; QIAO, Y.; XIANG, T.; LOY, C. C. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 45, p. 4396–4415, 2021. Disponível em: <https://api.semanticscholar.org/CorpusID:232104764>.

ZHOU, K.; YANG, Y.; HOSPEDALES, T.; XIANG, T. Learning to generate novel domains for domain generalization. In: VEDALDI, A.; BISCHOF, H.; BROX, T.; FRAHM, J.-M. (Ed.). *Computer Vision – ECCV 2020*. Cham: Springer International Publishing, 2020. p. 561–578. ISBN 978-3-030-58517-4.

ZHOU, L.; XUAN, O.; YING, H.; HAN, L.; CHENG, Y.; ZHANG, T. Cyber-attack classification in smart grid via deep neural network. In: *International Conference on Computer Science and Application Engineering*. [s.n.], 2018. Disponível em: <https://api.semanticscholar.org/CorpusID:53040178>.

ANEXO A – TABELAS DAS MÉTRICAS RELACIONADAS A IMPERCEPTIBILIDADE

Embora alguns valores de Ł$_2$/Ł$_\infty$ perturbados sejam menores que o original, no presente caso isso pode ser causado por ataques que otimizam a procura por perturbações mínimas, principalmente nos casos: CW , HopSkipJump e SignOPT de acordo com seus respectivos autores [Carlini e Wagner 2017], [Chen e Jordan 2019] e [Cheng et al. 2019], embora possa acontecer em ataques iterativos [Madry et al. 2017]. Para ter uma noção global da imperceptibilidade da amostra, as métricas foram calculadas para a amostra e não apenas para as features perturbadas.

A.1 CIC IDS2017

A.1.1 White-Box

Tabela 29 – Métricas para CIC IDS2017 e ataque PGD-100 contra classificadores binários

PGD	**Métrica**	**Epsilon**					
		0.1		***0.2***		***0.3***	
		Original	**Perturbed**	**Original**	**Perturbed**	**Original**	**Perturbed**
CLEAN	L_∞	0,99	1,00	0,99	1,00	0,99	1,00
	Wasserstein	0,00	0,01	0,00	0,02	0,00	0,03
IDS	L_∞	0,99	0,99	0,99	0,99	0,99	1,00
	Wasserstein	0,00	0,01	0,00	0,02	0,00	0,03

Tabela 30 – Métricas para CIC IDS2017 e ataque CW-10 contra classificadores binários

CW	**Métrica**	**Confidence**					
		0,00		***0,20***		***0,50***	
		Original	**Perturbed**	**Original**	**Perturbed**	**Original**	**Perturbed**
CLEAN	L_2	4,34	4,34	4,34	4,34	4,34	4,34
	Wasserstein	0,00	0,00	0,00	0,00	0,00	0,00
IDS	L_2	4,34	4,34	4,34	4,34	4,34	4,34
	Wasserstein	0,00	0,00	0,00	0,00	0,00	0,00

Tabela 31 – Métricas para CIC IDS2017 e ataque PGD-100 contra classificadores multiclasse

PGD	Métrica	Epsilon					
		0.1		*0.2*		*0.3*	
		Original	Perturbed	Original	Perturbed	Original	Perturbed
CLEAN	L_∞	0,99	0,99	0,99	1,00	0,99	1,00
	Wasserstein	0,00	0,01	0,00	0,02	0,00	0,03
IDS	L_∞	0,99	0,99	0,99	1,00	0,99	1,00
	Wasserstein	0,00	0,01	0,00	0,01	0,00	0,02

Tabela 32 – Métricas para CIC IDS2017 e ataque CW-10 contra classificadores multiclasse

CW	Métrica	Confidence					
		0,00		*0,20*		*0,50*	
		Original	Perturbed	Original	Perturbed	Original	Perturbed
CLEAN	L_2	4,34	4,34	4,34	4,34	4,34	4,34
	Wasserstein	0,00	0,00	0,00	0,00	0,00	0,00
IDS	L_2	4,34	4,34	4,34	4,34	4,34	4,34
	Wasserstein	0,00	0,00	0,00	0,00	0,00	0,00

A.1.2 Black-Box

Tabela 33 – Métricas para CIC IDS2017 e ataque HopSkipJump contra classificadores binários

HopSkipJump	Métricas	Original	Perturbed
CLEAN	L_∞	0,99	0,99
	Wasserstein	0,00	0,00
IDS	L_∞	0,99	0,99
	Wasserstein	0,00	0,00

Tabela 34 – Métricas para CIC IDS2017 e ataque SignOPT contra classificadores binários

SignOPT	Métricas	Original	Perturbed
CLEAN	L_2	4,34	4,35
	Wasserstein	0,00	0,02
IDS	L_2	4,34	4,46
	Wasserstein	0,00	0,02

Tabela 35 – Métricas para CIC IDS2017 e ataque HopSkipJump contra classificadores multiclasse

HopSkipJump	Métrica	Original	Perturbed
CLEAN	L_∞	0,99	0,99
	Wasserstein	0,0	0,0
IDS	L_∞	4,3	4,3
	Wasserstein	0,0	0,0

Tabela 36 – Métricas para CIC IDS2017 e ataque SignOPT contra classificadores multiclasse

SignOPT	Métrica	Original	Perturbed
CLEAN	L_2	4,34	4,35
	Wasserstein	0,0	0,0
IDS	L_2	4,3	4,4
	Wasserstein	0,0	0,0

A.2 UNSW-NB15

A.2.1 White-Box

Tabela 37 – Métricas para UNSW-NB15 e ataque PGD-100 contra classificadores binários

PGD	Métricas	Epsilon					
		0.1		***0.2***		***0.3***	
		Original	**Perturbed**	**Original**	**Perturbed**	**Original**	**Perturbed**
CLEAN	L_∞	113,86	**113,78**	113,86	**113,69**	113,86	**113,61**
	Wasserstein	0,00	0,02	0,00	0,04	0,00	0,05
IDS	L_∞	113,86	113,88	113,86	113,90	113,86	113,92
	Wasserstein	0,00	0,02	0,00	0,04	0,00	0,05

Tabela 38 – Métricas para UNSW-NB15 e ataque CW-10 contra classificadores binários

CW	Métricas	Confidence					
		0,00		***0,20***		***0,50***	
		Original	**Perturbed**	**Original**	**Perturbed**	**Original**	**Perturbed**
CLEAN	L_2	114,12	**114,08**	114,12	**114,08**	114,12	**114,08**
	Wasserstein	0,00	0,03	0,00	0,03	0,00	0,03
IDS	L_2	114,12	114,12	114,12	114,12	114,12	114,12
	Wasserstein	0,00	0,00	0,00	0,00	0,00	0,00

Tabela 39 – Métricas para UNSW-NB15 e ataque PGD-100 contra classificadores multiclasse

PGD	**Métricas**	**Epsilon**					
		0.1		***0.2***		***0.3***	
		Original	**Perturbed**	**Original**	**Perturbed**	**Original**	**Perturbed**
CLEAN	L_∞	11,24	**11,24**	11,24	**11,23**	11,24	**11,21**
	Wasserstein	0,00	0,02	0,00	0,04	0,00	0,06
IDS	L_∞	11,24	11,24	11,24	11,24	11,24	**11,22**
	Wasserstein	0,00	0,02	0,00	0,03	0,00	0,05

Tabela 40 – Métricas para UNSW-NB15 e ataque CW-10 contra classificadores multiclasse

CW	**Métricas**	**Confidence**					
		0,00		***0,20***		***0,50***	
		Original	**Perturbed**	**Original**	**Perturbed**	**Original**	**Perturbed**
CLEAN	L_2	13,40	13,40	13,40	13,40	13,40	13,40
	Wasserstein	0,00	0,00	0,00	0,00	0,00	0,00
IDS	L_2	13,40	13,40	13,40	13,40	13,40	13,40
	Wasserstein	0,00	0,00	0,00	0,00	0,00	0,00

A.2.2 Black-Box

Tabela 41 – Métricas para UNSW-NB15 e ataque HopSkipJump contra classificadores binários

HopSkipJump	**Métrica**	**Original**	**Perturbed**
CLEAN	L_∞	10,27	**10,23**
	Wasserstein	0,00	0,10
IDS	L_∞	10,27	10,27
	Wasserstein	0,00	0,09

Tabela 42 – Métricas para UNSW-NB15 e ataque SignOPT contra classificadores binários

SignOPT	**Métrica**	**Original**	**Perturbed**
CLEAN	L_2	12,60	12,76
	Wasserstein	0,00	0,07
IDS	L_2	12,60	12,75
	Wasserstein	0,00	0,06

Tabela 43 – Métricas para UNSW-NB15 e ataque HopSkipJump contra classificadores multiclasse

HopSkipJump	Métrica	Original	Perturbed
CLEAN	L_∞	11,24	**11,14**
	Wasserstein	0,00	0,17
IDS	L_∞	11,24	**11,21**
	Wasserstein	0,00	0,07

Tabela 44 – Métricas para UNSW-NB15 e ataque SignOPT contra classificadores multiclasse

SignOPT	Métrica	Original	Perturbed
CLEAN	L_2	13,40	13,53
	Wasserstein	0,00	0,07
IDS	L_2	13,40	13,41
	Wasserstein	0,00	0,02

ANEXO B – ATAQUES DE REDE E MITRE ATT&CK

Tabela 45 – Resumo dos ataques Reconnaissance, Generic, DoS e DDoS com descrições em tópicos

Ataque	**Descrição**	**Objetivo**	**Exemplos**	**Danos/Prejuízos (MITRE ATT&CK)**
Recon.	▪ Coleta de informações sobre o alvo ▪ Infraestrutura, pessoas e sistemas	▪ Mapear superfície de ataque ▪ Identificar vulnerabilidades ▪ Preparar etapas futuras	▪ Scans de rede ▪ Coleta de e-mails ▪ Pesquisa em redes sociais ▪ Consulta a bancos de dados públicos	▪ Exposição de informações sensíveis ▪ Aumento do risco de ataques direcionados ▪ Preparação para ataques subsequentes
Generic	▪ Técnica aplicável a qualquer cifra de bloco ▪ Não explora detalhes do algoritmo	▪ Quebrar criptografia por métodos universais ▪ Força bruta ou colisão	▪ Força bruta de chaves ▪ Ataque de aniversário (birthday attack)	▪ Quebra de confidencialidade ▪ Acesso não autorizado a informações protegidas ▪ Exposição de credenciais
DoS	▪ Interrupção ou degradação de serviço ▪ Sobrecarga ou exploração de falhas	▪ Tornar serviço indisponível para usuários legítimos	▪ Sobrecarga de recursos ▪ Envio de pacotes malformados ▪ Crash persistente	▪ Indisponibilidade de serviços ▪ Interrupção de operações críticas ▪ Dano reputacional e financeiro
DDoS	▪ DoS realizado por múltiplos sistemas distribuídos ▪ Usualmente via botnets	▪ Esgotar largura de banda ou recursos do alvo ▪ Ataque massivo e coordenado	▪ SYN flood ▪ UDP flood ▪ HTTP flood ▪ Tráfego volumétrico distribuído	▪ Indisponibilidade total ou parcial de serviços ▪ Perda de receita ▪ Impacto em clientes ▪ Distração para outros ataques

OBS: As definições dos ataques usadas nesta tabela foram retiradas da documentação dos datasets. As demais descrições foram adaptadas do MITRE ATT&CK.

ANEXO C – RESULTADOS PARA TREINOS ADVERSARIAIS EM DADOS DESBALANCEADOS

C.1 COMPARATIVO ENTRE OS TREINOS

A tabela 46 abaixo mostra os resultados para classificação multi-classe com o treino adversarial usado neste trabalho com a função de custo utilizada (Entropia cruzada). Uma vez que o treinamento adversarial mostra problemas com dados desbalanceados Li, Xu e Zhang 2023, foi tentado as seguintes abordagens de forma obter um modelo que não tivesse performance ruim nas classes minoritárias:

- MAIL: Wang et al. 2021 propuseram uma estratégia adaptativa de associar os pesos de classes na função entropia cruzada. Resumidamente conforme uma certa quantidade de épocas o método muda um fator que é multiplicado pelo valor do custo das classes.
- LDAM+SCL: Em Wang et al. 2021 o qual usam duas funções de custo combinadas: LDAM Cao et al. 2019 e contrastiva supervisionada (SCL) Khosla et al. 2020. Segundo os autores,esta combinação de funções de custo conseguiriam ponderar de maneira eficiente o peso das classes durante o treino.

Não foram realizados experimentos com LDAM+SCL ou MAIL para CIC IDS2017 uma vez que os resultados obtidos com o UNSW-NB15 não foram satisfatórios.

Tabela 46 – Entropia cruzada VS MAIL VS LDAM+SCL em treino adversarial (MPB-AT)

UNSW-NB15									
Label	**Entropia Cruzada**			**MAIL**			**LDAM+SCL**		
	Precision	***Recall***	***ROC-AUC***	***Precision***	***Recall***	***ROC-AUC***	***Precision***	***Recall***	***ROC-AUC***
Normal	100	99,52	99,88	99,38	99,83	97,69	100	42,52	93,52
Exploits	71,2	61,46	98,28	55,33	58,5	94,86	1,59	24,68	68,76
Recon.	82,86	92,05	98,65	87,16	36,65	98,58	5,83	89,49	97,27
DoS	20,53	23,18	84,37	8,4	4,72	91,35	0,21	38,2	61,6
Generic	96,85	96,01	99,17	94,34	76,53	90,36	52,6	91,36	99
Shellcode	41,76	84,44	95,91	0	0	89,25	0,99	93,33	99,18
Fuzzers	53,45	77,52	99,73	40,96	33,66	97,16	4,63	63,56	86,17
Worms	8,33	20	41	0	0	99,35	0,04	100	96,36
Backdoor	12,5	35,51	97,57	0	0	75,63	0,13	31,78	64,32
Analysis	12,82	61,9	99,83	0	0	85,36	3,51	69,52	96,52

ANEXO D – GRÁFICOS RECALL E ROC-AUC

D.1 RECALL - CIC IDS2017 - WHITE BOX

Figura 20 – Recall após PGD-100 com CIC IDS2017 e classificadores binários

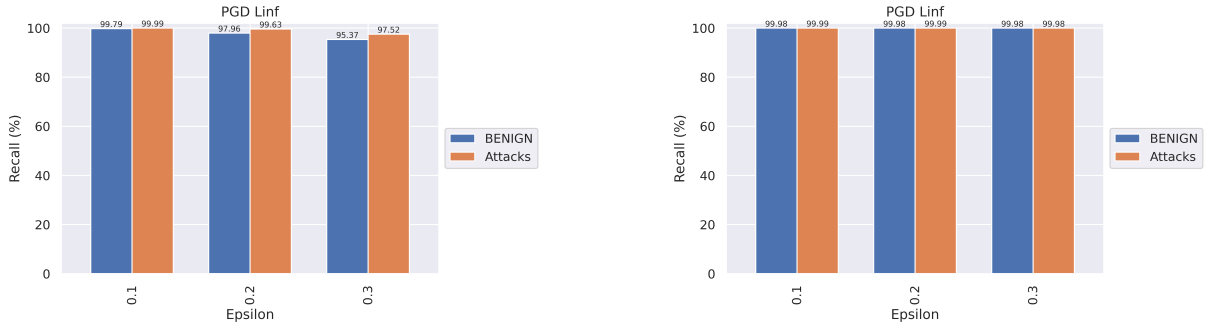


Figura 21 – Recall após CW-10 com CIC IDS2017 e classificadores binários

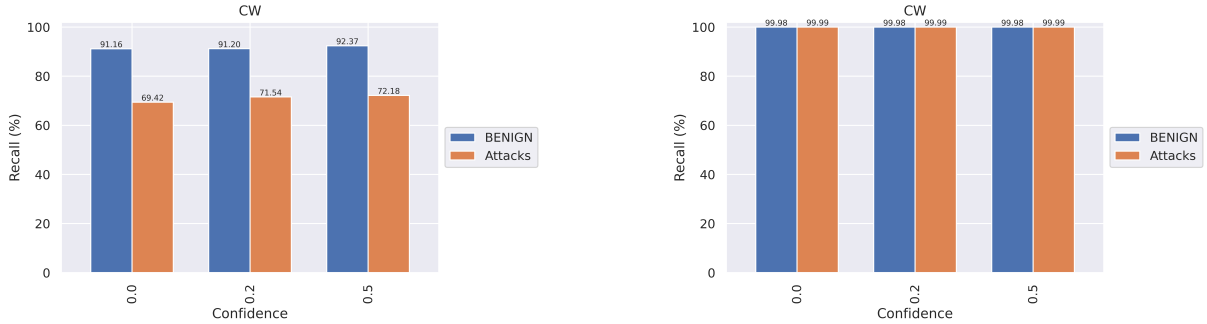


Figura 22 – Recall após PGD-100 com CIC IDS2017 e classificadores multiclasse

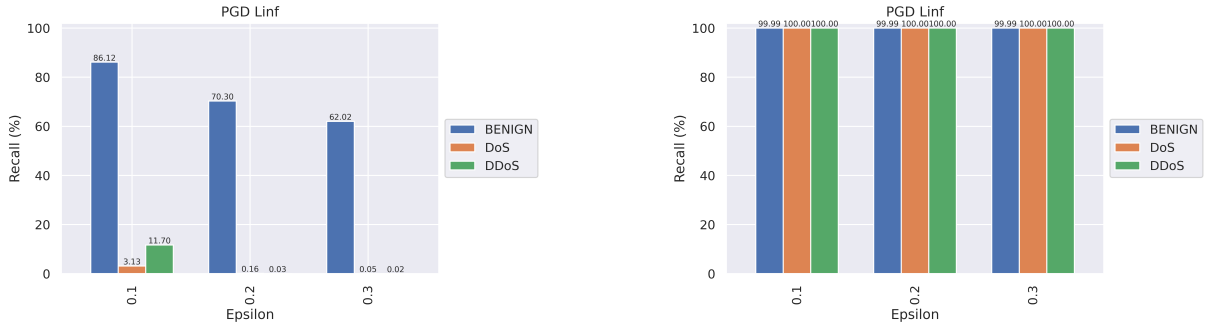
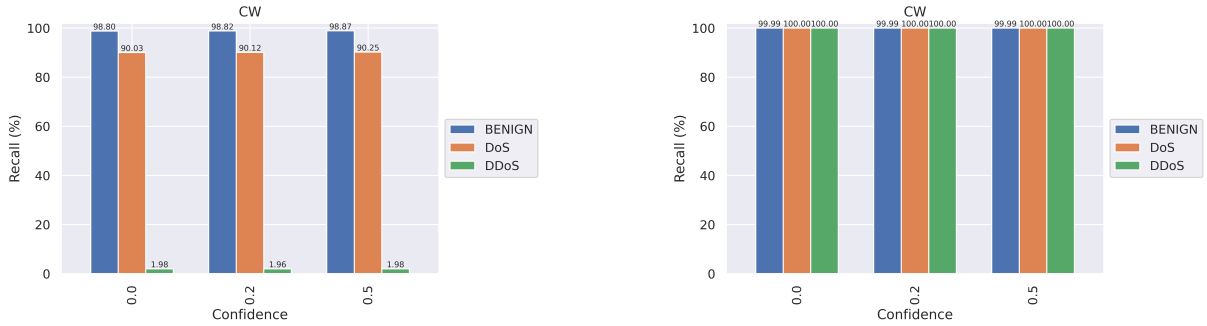


Figura 23 – Recall após CW-10 com CIC IDS2017 e classificadores multiclasse



D.2 RECALL - CIC IDS2017 - BLACK BOX

Figura 24 – Recall após HopSkipJump com CIC IDS2017 e classificadores binários

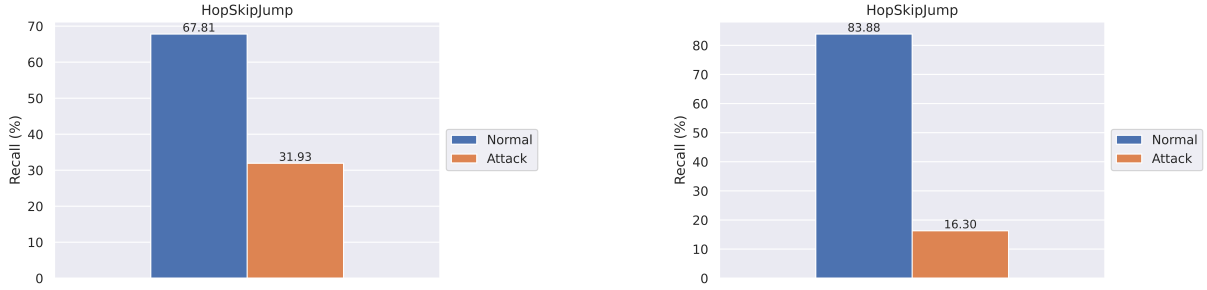


Figura 25 – Recall após SignOPT com CIC IDS2017 e classificadores binários

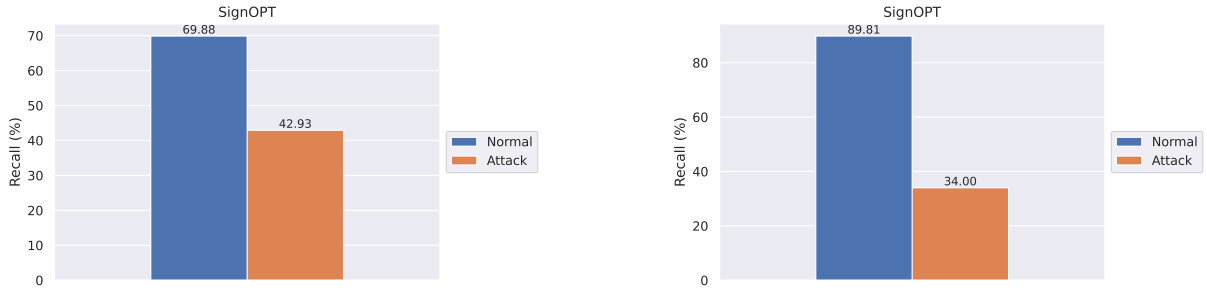


Figura 26 – Recall após HopSkipJump com CIC IDS2017 e classificadores multiclasse

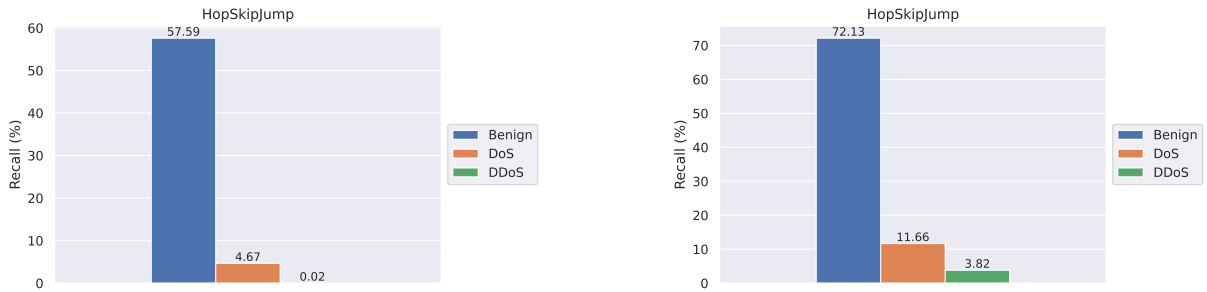
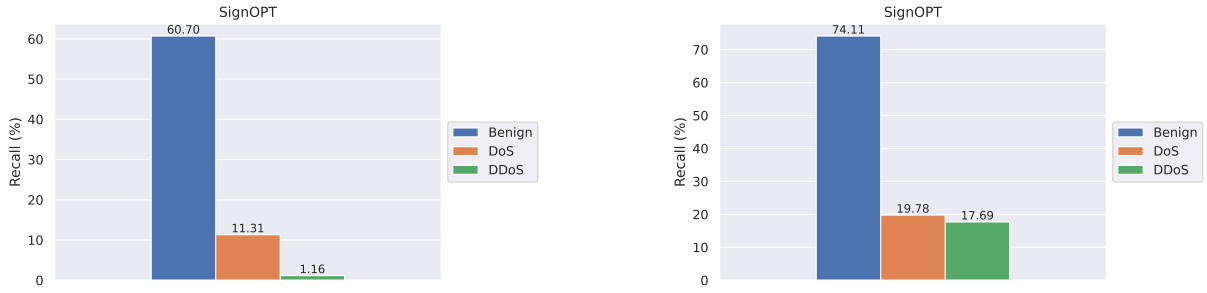


Figura 27 – Recall após SignOPT com CIC IDS2017 e classificadores multiclasse



D.3 RECALL - UNSW-NB15 - WHITE BOX

Figura 28 – Recall após PGD-100 com UNSW-NB15 e classificadores binários

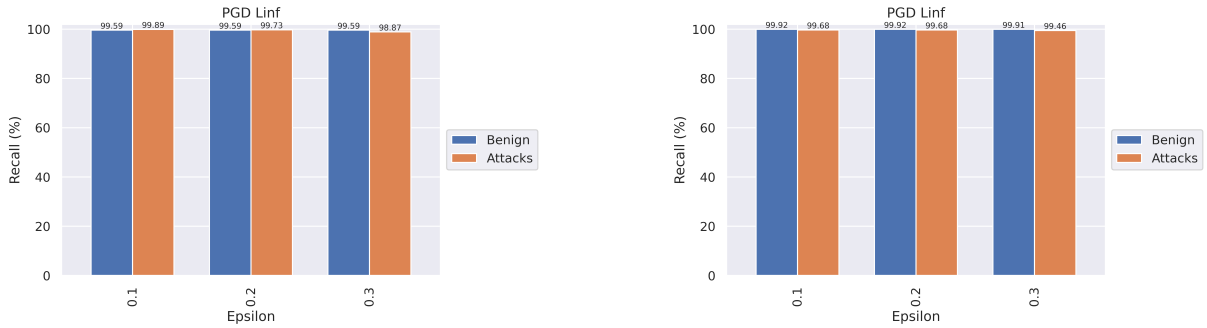


Figura 29 – Recall após CW-10 com UNSW-NB15 e classificadores binários

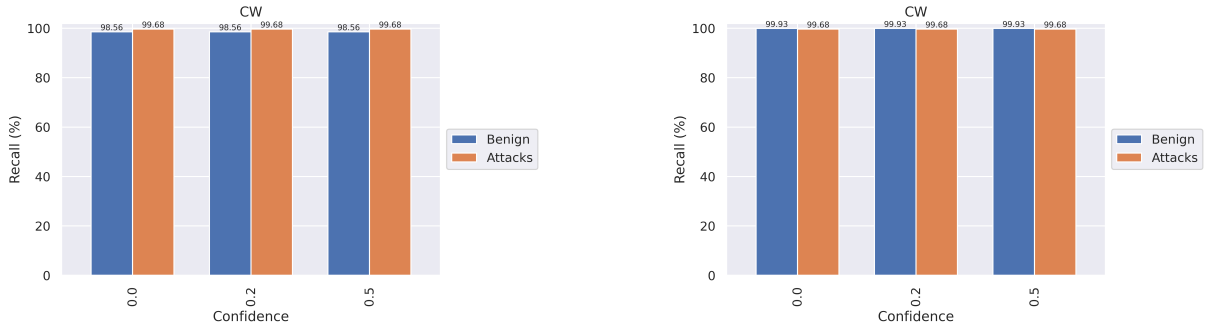


Figura 30 – Recall após PGD-100 com UNSW-NB15 e classificadores multiclasse

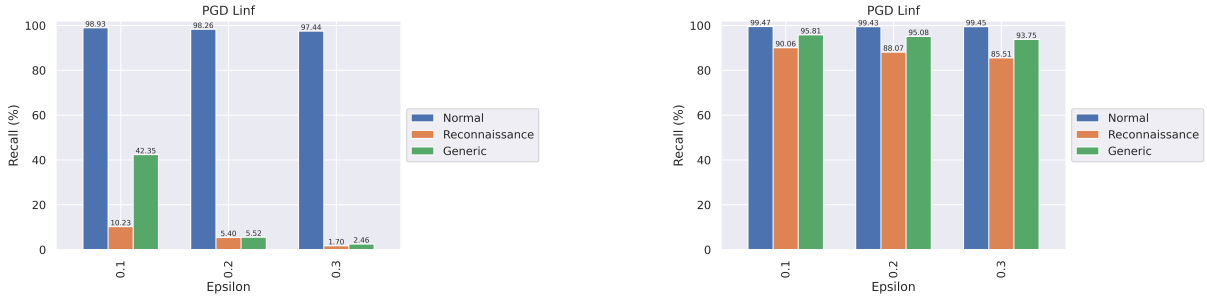
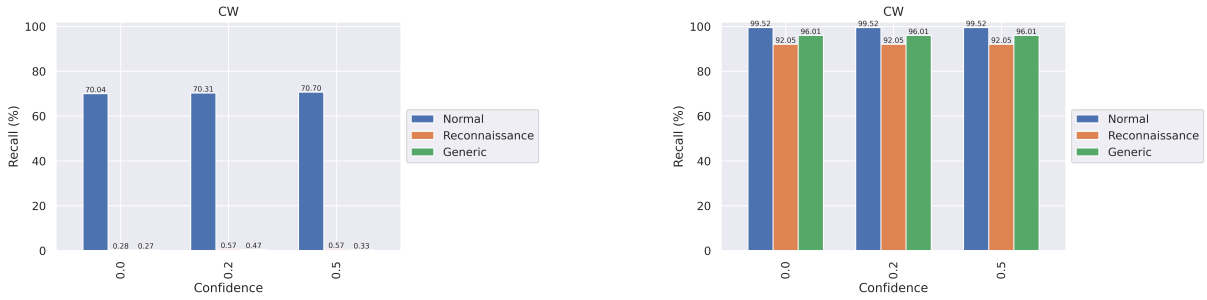


Figura 31 – Recall após CW-10 com UNSW-NB15 e classificadores multiclasse



D.4 RECALL - UNSW-NB15 - BLACK BOX

Figura 32 – Recall após HopSkipJump com UNSW-NB15 e classificadores binários

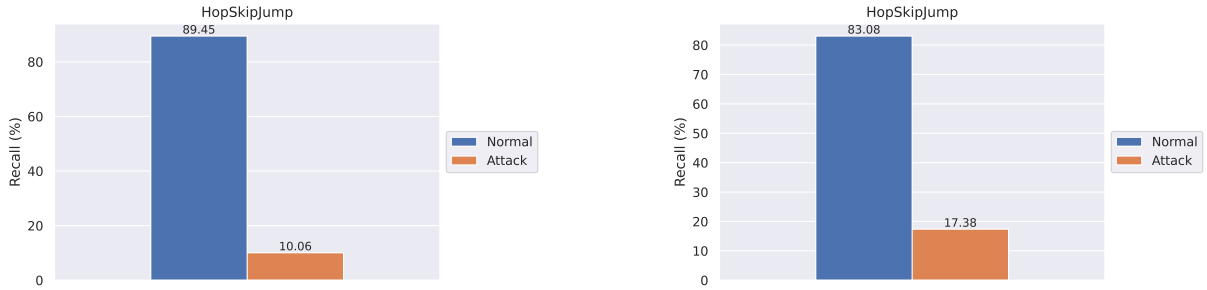


Figura 33 – Recall após SignOPT com UNSW-NB15 e classificadores binários

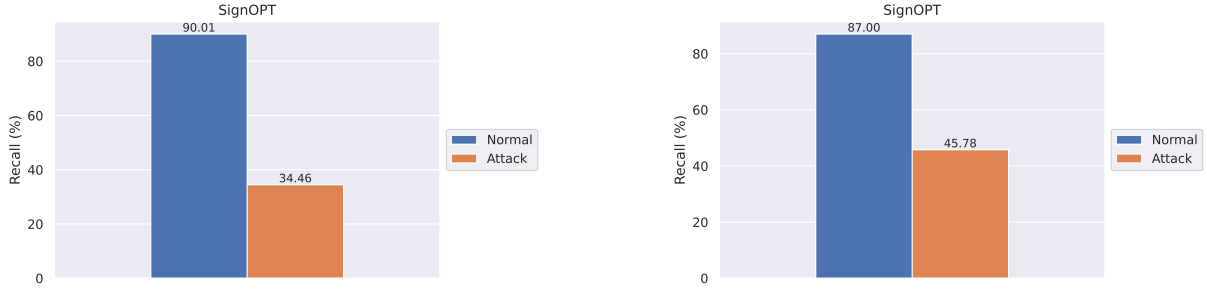


Figura 34 – Recall após HopSkipJump com UNSW-NB15 e classificadores multiclasse

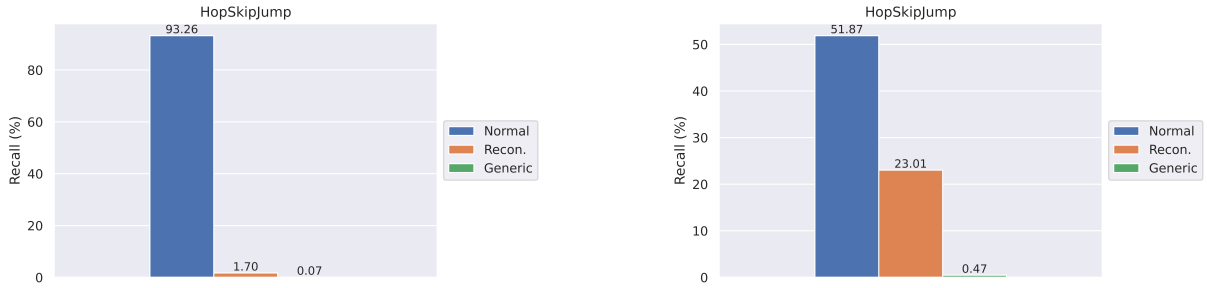
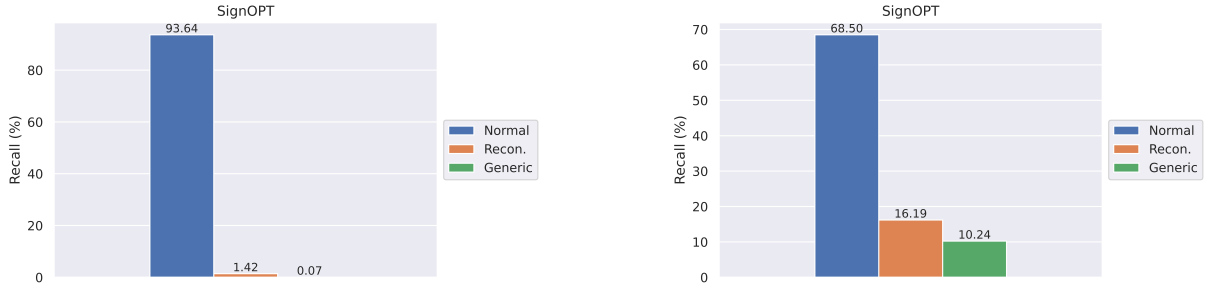


Figura 35 – Recall após SignOPT com CIC IDS2017 e classificadores multiclasse



D.5 ROC-AUC - CIC IDS2017 - WHITE BOX

Figura 36 – ROC-AUC após PGD-100 com CIC IDS2017 e classificadores binários

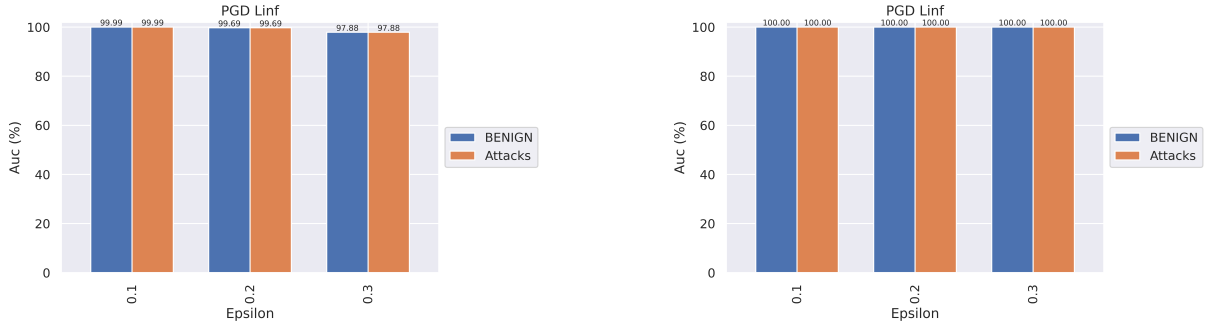


Figura 37 – ROC-AUC após CW-10 com CIC IDS2017 e classificadores binários

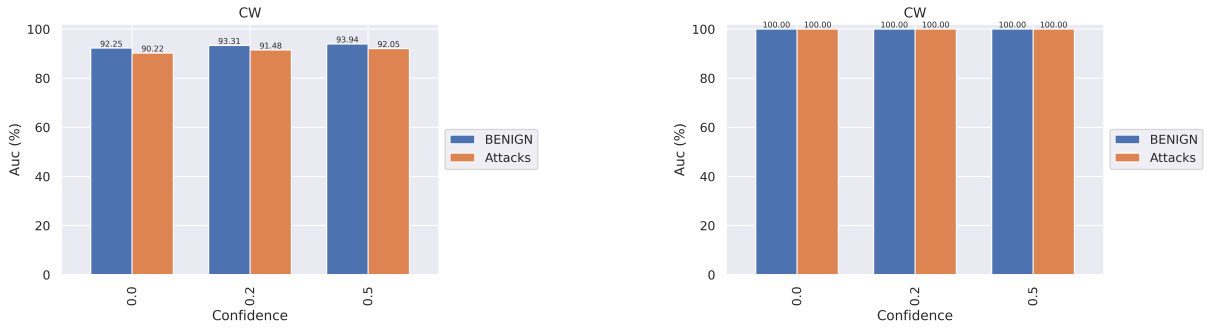


Figura 38 – ROC-AUC após PGD-100 com CIC IDS2017 e classificadores multiclasse

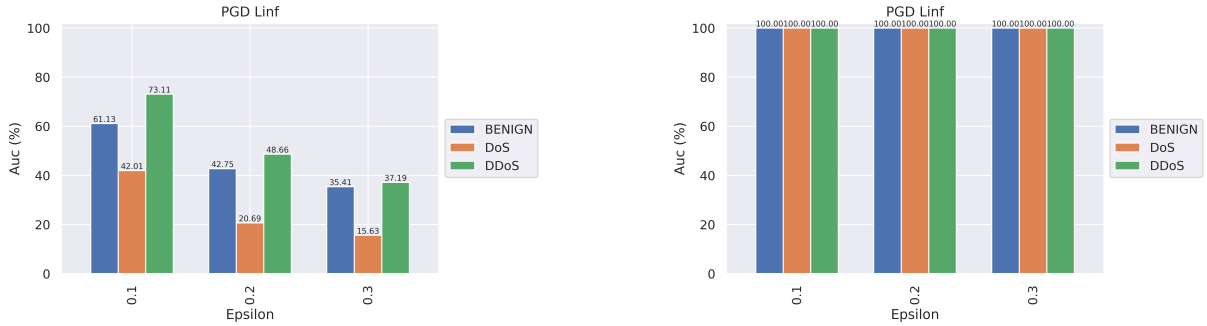
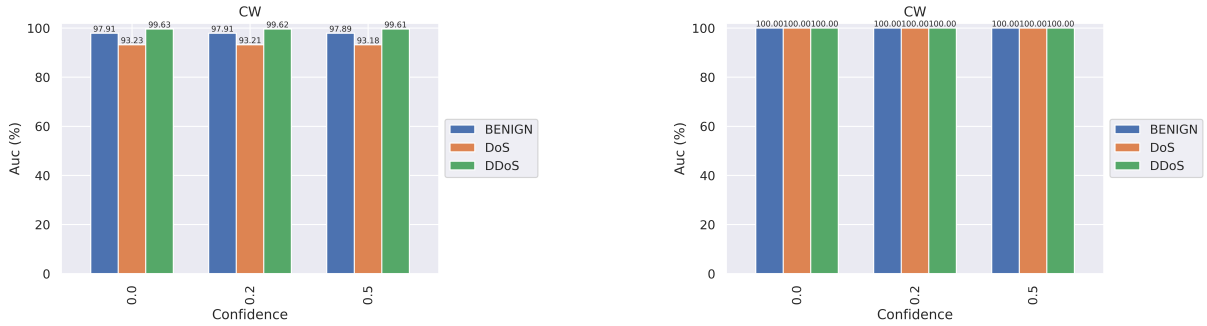


Figura 39 – ROC-AUC após CW-10 com CIC IDS2017 e classificadores multiclasse



D.6 ROC-AUC - CIC IDS2017 - BLACK BOX

Figura 40 – ROC-AUC após HopSkipJump com CIC IDS2017 e classificadores binários

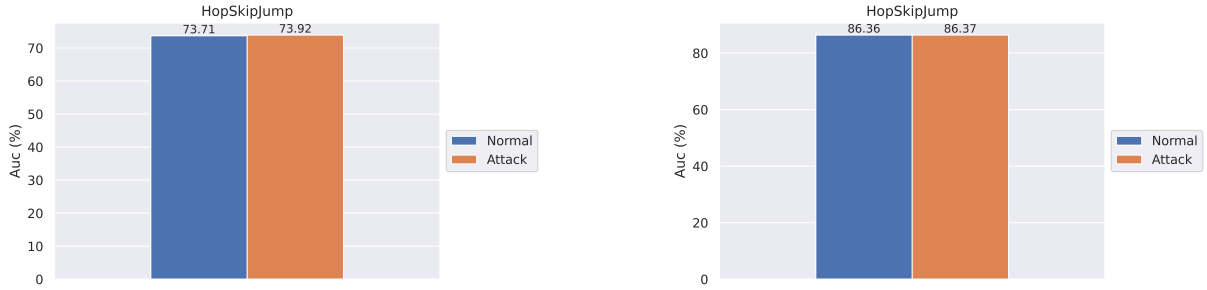


Figura 41 – Recall após SignOPT com CIC IDS2017 e classificadores binários

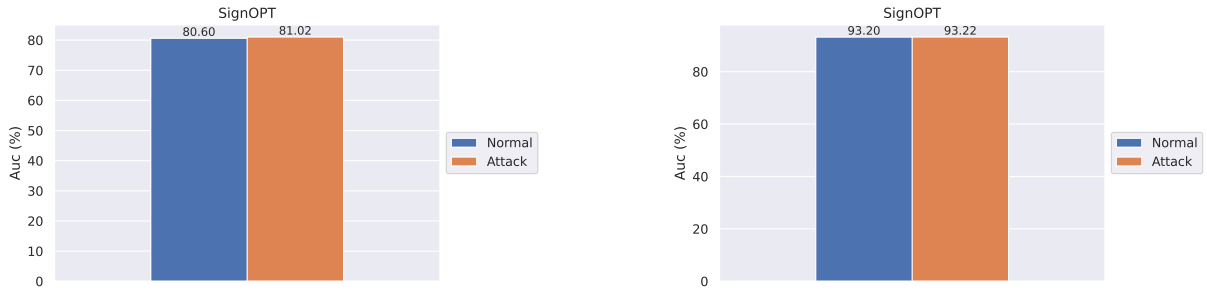


Figura 42 – ROC-AUC após HopSkipJump com CIC IDS2017 e classificadores multiclasse

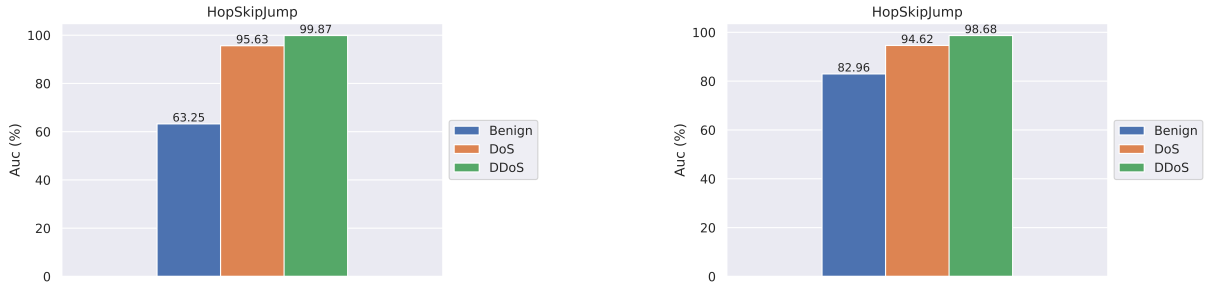
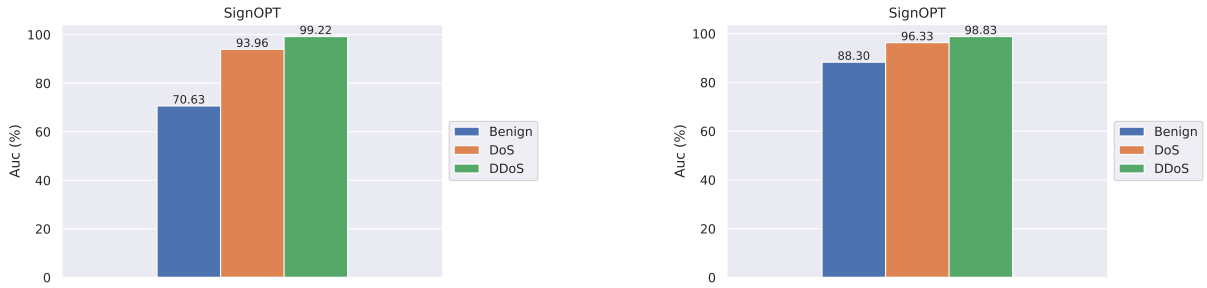


Figura 43 – ROC-AUC após SignOPT com CIC IDS2017 e classificadores multiclasse



D.7 ROC-AUC - UNSW-NB15 - WHITE BOX

Figura 44 – ROC-AUC após PGD-100 com UNSW-NB15 e classificadores binários

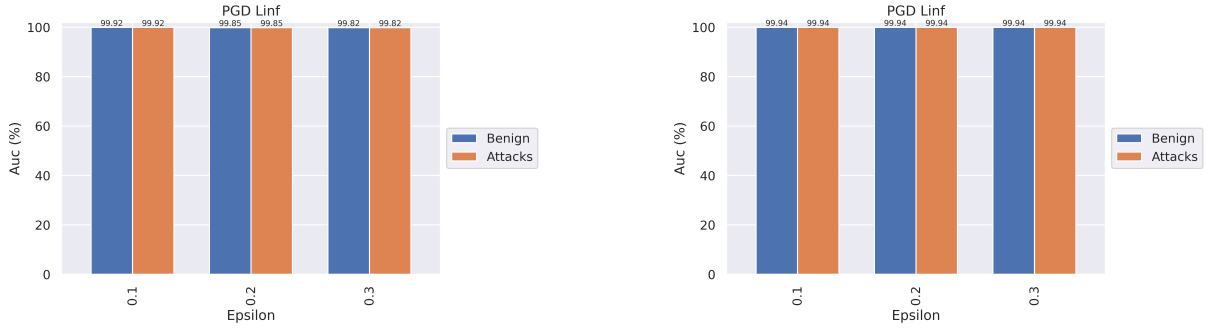


Figura 45 – ROC-AUC após CW-10 com UNSW-NB15 e classificadores binários

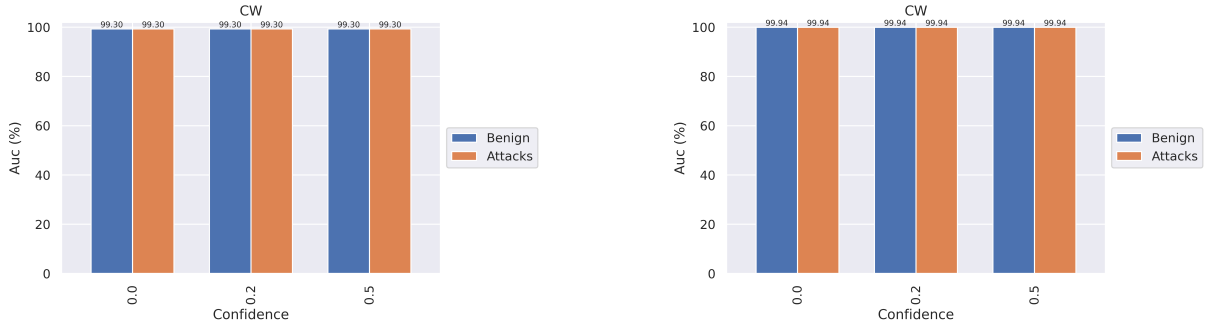


Figura 46 – ROC-AUC após PGD-100 com UNSW-NB15 e classificadores multiclasse

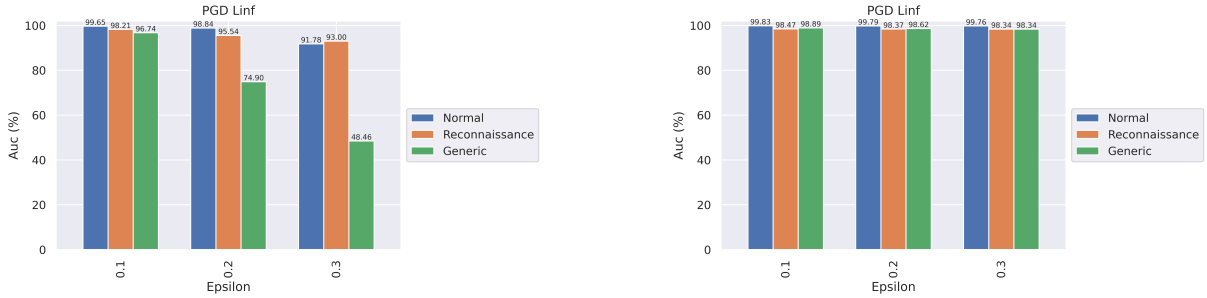
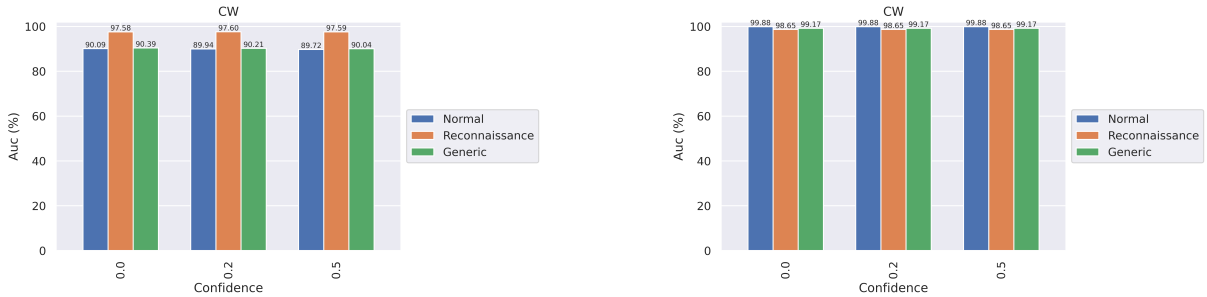


Figura 47 – ROC-AUC após CW-10 com UNSW-NB15 e classificadores binários



D.8 ROC-AUC - UNSW-NB15 - BLACK BOX

Figura 48 – ROC-AUC após HopSkipJump com UNSW-NB15 e classificadores binários

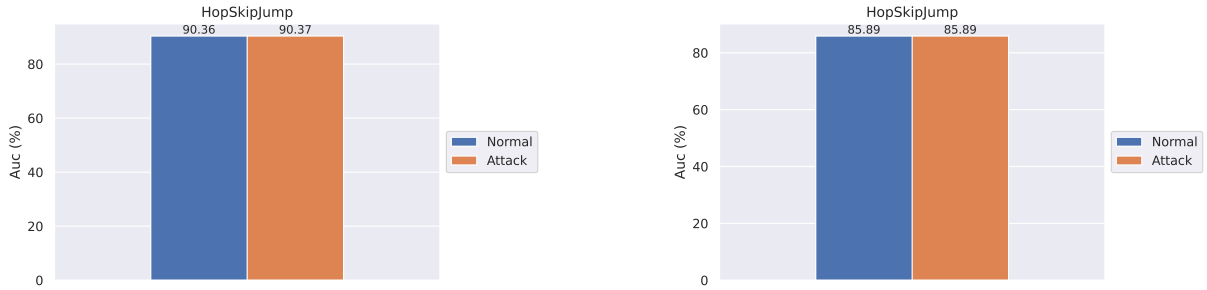


Figura 49 – Recall após SignOPT com UNSW-NB15 e classificadores binários

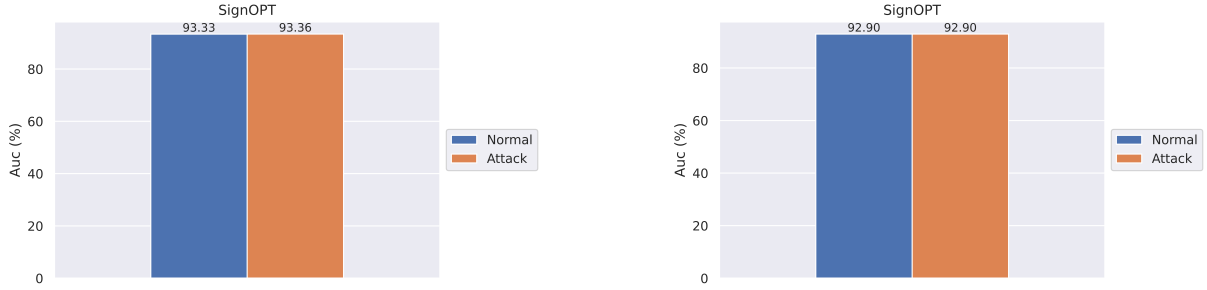


Figura 50 – ROC-AUC após HopSkipJump com UNSW-NB15 e classificadores multiclasse

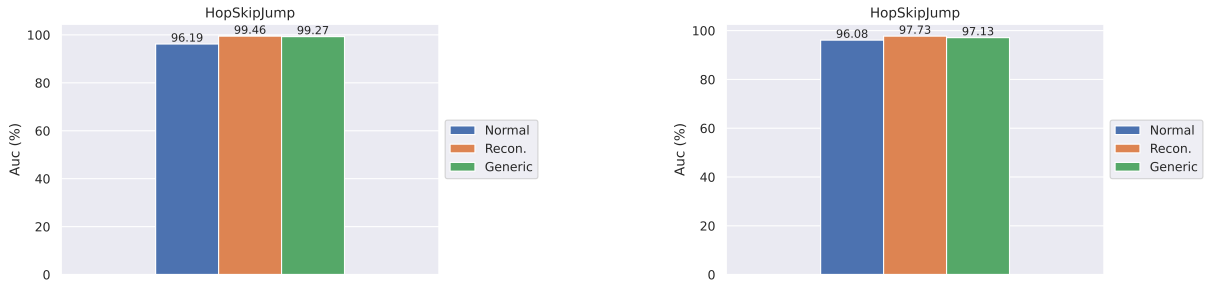


Figura 51 – ROC-AUC após SignOPT com UNSW-NB15 e classificadores multiclasse